INRIA
CENTRE DE ROCQUENCOURT

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél. : (1) 39 63 55 11

Rapports de Recherche

N° 342

DIAMOND SCHEMES : AN ORGANIZATION OF PARALLEL MEMORIES FOR EFFICIENT ARRAY PROCESSING

William JALBY
Jean-Marc FRAILONG
Jacques LENFANT

Octobre 1984

DIAMOND SCHEMES: AN ORGANIZATION OF PARALLEL MEMORIES FOR EFFICIENT ARRAY PROCESSING

WILLIAM JALBY

INRIA, Rocquencourt, France

JEAN-MARC FRAILONG

Ecole Normale Superieure/CNRS, Paris, France

JACQUES LENFANT

IRISA/Universite de Rennes, Rennes, France

RESUME : Sur un ordinateur vectoriel, l'organisation des tableaux de données en mémoire, conditionne l'éfficacité des accés mémoire et par là la performance globale de la machine. Ce probleme peut être formulé en termes de compatibilité entre une fonction de rangement et une famille de fenêtres. Une fonction de rangement est une fonction associant à chaque élement d'un tableau le numéro du banc mémoire dans lequel il est stocké; une fenêtre est un ensemble d'éléments d'un tableau devant être accédés simultanément: partie de ligne, colonne, diagonale, bloc carré,... La première analyse en profondeur de ce problème a été réalisée par Shapiro. Son travail repose sur une notion de compatibilité rendant équivalent l'étude de l'accés en paralléle d'une seule fenêtre et celle de l'accés en paralléle à tous les translatées de cette fenêtre. En pratique, cette hypothèse s'avère beaucoup trop contraignante car seul un sous-ensemble restreint des translatées d'une fenêtre est utile. La théorie de Shapiro a pour défaut majeur que dans nombre de cas elle ne permet pas de trouver une fonction de rangement compatible avec les fenêtres requises. Dans cet exposé, nous utiliserons une définition de la compatibilité permettant de privilégier la forme géométrique des fenêtres par rapport au nombre de translatées. Dans ce but, nous analyserons les propriétés d'une nouvelle classe de fonctions de rangement: les schémas en diamant. Enfin nous donnerons des exemples d'utilisation de ces schémas pour des methodes multigrilles et des traitements d'images.

ABSTRACT : In a parallel vector computer, the way of storing arrays conditions the efficiency of memory accesses. This recognized problem may have a dramatic effect on the overall performance. It can be stated in terms of compatibility between a skewing scheme and a family of access templates. A skewing scheme is a function expressing in which memory bank each array element is stored whereas an access template is the "form" of of a set of array elements to be accessed simultaneously: portion of row, column or diagonal, square block,... The first in depth analysis of the parallel storage problem is Shapiro's work. It is based on a concept of compatibility assuming that, whatever the translation applied to the reference access template, the array elements within its image are stored in distinct banks. This requirement is unnecessarily constraining in many practical situations where not all translations need to be considered. As a troublesome consequence it frequently happens that no skewing scheme is compatible with all of a family of access templates desired required by a programmer. In this paper we modify the compatibility objective so that the variety of access templates may prevail over the density of possible translations from the reference position. From this point of view we analyze the structure of diamond schemes, a new class of skewing schemes derived from a lattice of permutations. As examples we suggest array implementations for multigrid methods and for image processing.

Categories and Subject Descriptors : B.3.2 [Memory Structures]: Design styles - Interleaved memories; C.1.2 [Processor Architectures]: Multiple Data Stream Architectures - Array and vector processors; E.2 [Data Storage Representations]: Contiguous representations; G.4 [Mathematical Software]: Algorithm analysis - Efficiency.

General terms : design, performance, theory.

Additional Keywords and Phrases : skewing scheme, multigrid method, image processing.

Author's addresses

W.Jalby: INRIA, B.P.105, 78153 Le Chesnay Cedex, France.

J.-M.Frailong: Laboratoire d'Informatique, Ecole Normale Superieure, 45 rue d'Ulm, 75230 Paris Cedex 05, France.

J.Lenfant: IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France.

1. INTRODUCTION

One of the main issues in supercomputer architecture is the design of a main memory having a bandwidth fitting the throughput of the arithmetic operators. SIMD architectures answer this problem by using fully parallel access to memory, thus providing computationnal units with a whole data vector at a time. The effective memory bandwidth will reach its peak only if the data to be accessed lie in different memory modules. This implies that data must be organized in the memory in such a way that all accesses required by a given algorithm are conflict-free, i.e. can be made in parallel. It also implies that data must be unscrambled between memory and processors, so that operands required by a computational unit are brought together correctly.

The usual solutions to the parallel data access problem in SIMD machines imply dissymetric architectures, with a number of memory modules larger than the number of processing elements. The latter is usually a power of two, for the sake of hardware design simplification. A first approach is to use a number of memory banks that is a prime number sligthly larger than the number of processing elements [5]. Such is the solution used in BSP [6] [7] [8], and suggested by BURROUGHS for the NASF project. This organization gives easy access to rows, columns and diagonals of matrices. The drawback of this solution is that it requires non-symmetric networks, address computation modulo the number of memory banks, and that it gives little opportunity to access other types of templates. A second approach, proposed by Lawrie [9], uses twice as many memory modules as processors. It is thus rather expensive, due to the number of memory modules and to the increase in size of the interconnection network required.

In this paper, we will use a minimal hardware configuration, with N processors, N memory banks, and an $N \times N$ interconnection network (Fig.1). For this architecture, we will develop techniques for data skewing ensuring maximum effective memory bandwidth for a given algorithm. In all our examples, N will be a power of two; although the theory developped here does not depend on it, it is the most interesting case from the hardware designer's point of view as it corresponds to the size of the most eficient networks, such as the Ω, baseline and Benes networks [10] [20].

For a given algorithm, computations can be organized to modify the type of parallel access required. Let us take as an example the computation of the discretization of the laplacian Δf on a finite difference grid. For each point (i,j), Δf is given by:

$$\Delta f(i,j) = \frac{1}{4} [f(i+1,j) + f(i-1,j) + f(i,j+1) + f(i,j-1) - 4\ f(i,j)]$$

On a parallel machine, two approaches are possible to compute Δf. The first one consists in emphasizing the parallel access requirements of the pointwise equation. Then, we will need to access in parallel, for each point (i,j), f(i,j) and its four neighbours in the north, east, south and west directions. This means that the array f must be organized so that it allows parallel access to a cross-shaped template centered at any point. This approach has been developped by Shapiro [3] [4].

In a second approach, we emphasize the fact that the computations at each point (i,j) are independant. Taking into account the degree of parallelism of the machine, we will partition the grid into subsets and compute Δf in parallel over all points of a subset. Then, for each subset S, we will perform consecutively the following parallel memory accesses:

- access to S, giving f(i,j)

- access to $S + (1,0)$, giving $f(i+1,j)$
- access to $S + (-1,0)$, giving $f(i-1,j)$
- access to $S + (0,1)$, giving $f(i,j+1)$
- access to $S + (0,-1)$, giving $f(i,j-1)$

This means that f must be organized to allow parallel access only to the subsets S and to the templates obtained by a translation of a subset S by one of the vectors (1,0), (-1,0), (0,1), (0,-1).

For an SIMD machine, the second approach is the reasonable one, as it reflects the simultaneity between the processing elements. In fact, this approach is also the appropriate one for general purpose vector computers, which lack special-purpose hardware for sophisticated addressing.

The main theoretical results on parallel access to templates have been obtained by Shapiro [3] [4]. If T is a template in Z^2, there exists a skewing scheme giving parallel access to the patterns obtained by all translations over Z^2 of T iff T tesselates Z^2 by translations. The same kind of result holds for a finite set of templates; there exists a skewing scheme giving parallel access simultaneously to a family of templates $T_1, \ldots, T_n$ and their images by all the translations over Z^2 iff each T_i tesselates the plane with the same set of translations. These theorems, which fit in the first approach described earlier, put very restrictive constraints on the patterns. For the second approach, they could also be used, but it is tempting to weaken the conditions of the theorem.

By noticing that only certain translations are really required by the algorithm, we will build skewing schemes giving parallel access to families of templates that cannot be covered by Shapiro's results. For example, it is possible to access in parallel rows, columns, and all possible rectangular $2^p \times 2^q$ templates in a matrix, for a subset of translations that still provide a full cover of the whole matrix for each template. Shapiro has shown that, if all translations are needed, it is impossible to access in parallel rows, columns and a single rectangular template.

We will first define a class of well-behaved skewing schemes, called diamond schemes, that includes most of the skewing schemes found in the literature and we will study their basic properties. Then, we will develop a few tools on permutations necessary to manipulate the mathematical structure efficiently. The following section will state the main results concerning compatibility of diamond schemes with data templates. Finally, we will describe three applications that show the interest of this class of skewing schemes.

2. PROBLEM FORMULATION AND DEFINITIONS

We consider the problem of skewing a matrix A into N memory modules. The matrix A is assumed to be infinite, indexed by Z^2. This does not change the nature of the problem and allows to neglect boundary conditions, which can be handled by processor masking.

Definition 2.1 : A skewing scheme is a mapping ϕ of Z^2 into $\{0..N-1\}$, where $\phi(i,j)$ is the number of the memory bank in which A_{ij} is stored [5].

In our figures, we will depart from the usual representations of matrices. $A_{i,j}$ will be represented as a point in Z^2 using the standard basis. The unit square based at point (i,j) will contain the value $\phi(i,j)$ of the skewing scheme.

Definition 2.2 : A data template is a finite subset of Z^2. It will be used as a parallel access window. A template T is compatible with the skewing scheme ϕ iff the restriction $\phi_{|T}$ of ϕ to T is injective, i.e. if all elements of T may be fetched in a single memory cycle.

It should be noted that, in Shapiro's formulation of the problem, a skewing scheme ϕ is valid for T iff, for all translations τ of Z^2, $\tau(T)$ is compatible with ϕ.

Definition 2.3 : The reference rectangle is a given rectangle defined by $R_{x,y} = \{ (i,j) \in Z^2 \,/\, 0 \leq i < x \,,\, 0 \leq j < y \}$ with $xy \leq N$. The reference covering of Z^2 is the set of all rectangles that are images of $R_{x,y}$ by the translations in the set Π :

$$\Pi = \{ \tau \in Z^2 / \tau = aX+bY \,,\, X = (x,0),\, Y = (0,y),\, (a,b) \in Z^2 \}$$

In order to simplify notations, we will denote the reference rectangle as R instead of $R_{x,y}$, except when ambiguous. Symbol $\pi_{a,b}$ will denote the translation aX+bY, an element of Π, and we will systematically identify $\pi_{a,b}$ with the point (aX,bY) in Z^2, and identify the groups (Π,o) and $(Z^2,+)$. Fig. 2 shows $R_{4,2}$ and the associated covering.

Greek letters λ, μ will denote permutations of {0..N-1}, π a translation in Π, τ any translation of Z^2.

If λ, μ are two permutations on {0..N-1} commuting with each other, i.e. $\lambda o \mu = \mu o \lambda$, and if π = aX+bY is a translation in Π, then we will note $[\lambda,\mu]^\pi$ for $\lambda^a \circ \mu^b$. It is obvious that:

$$[\lambda,\mu]^{\pi+\pi'} = [\lambda,\mu]^\pi \circ [\lambda,\mu]^{\pi'} . \qquad (1)$$

Our main tool to define skewing schemes is to extend a scheme from the reference rectangle to the reference covering.

<u>Definition 2.4</u> : Let λ and μ be two permutations of {0..N-1} commuting with each other and let φ be a mapping from R to {0..N-1}. The extension $\phi^{\lambda,\mu}$ of φ by (λ,μ) is defined over Z^2 by :

$$\text{for all } \pi \text{ in } \Pi, \quad \phi^{\lambda,\mu}_{|\pi(R)} = [\lambda,\mu]^\pi \circ \phi_{|R} \circ \pi^{-1} \qquad (2)$$

This definition is consistent, as the set { π(R) / π ∈ Π} covers Z^2 (it is our reference covering). Moreover, $\phi^{\lambda,\mu}_{|R} = \phi$. Whenever no ambiguity may arise, we will simply use φ instead of $\phi^{\lambda,\mu}$.

If $\phi^{\lambda,\mu}_{|R}$ is a one-to-one mapping onto {0..N-1}, $\phi^{\lambda,\mu}$ will be said to be a regular extension. This of course implies that xy = N.

Definition 2.5 : A skewing scheme ψ is a diamond scheme iff there exist a reference rectangle R, a mapping ϕ from R into {0..N-1} and two permutations on {0..N-1}, λ and μ, commuting with each other, such that $\psi = \phi^{\lambda,\mu}$.

Definition 2.6 : A skewing scheme ψ is a regular diamond scheme iff it is equal to some regular extension $\phi^{\lambda,\mu}$.

It is noteworthy that a diamond scheme may be obtained by several extensions. As shown by example 3 below, the reference rectangles of the various extensions yelding the same diamond scheme need not to have the same cardinality. Some of these extensions may be regular and others not.

If a skewing scheme ψ is a regular diamond scheme with reference rectangle R, it is compatible with all rectangles of the reference covering associated with R. Some other translations may belong to the set Γ of all permutations τ such that the regular diamond scheme is compatible with $\tau(R)$: $\Pi \subset \Gamma \subset Z^2$.

Shapiro [3][4] uses a similar mechanism to derive a skewing scheme from a tesselation of the plane. Although the translations operating on the base template to generate this tesselation are not required to form a group (as in Π), this condition is fulfilled in all applications. Extensions used by Shapiro rely on permutations λ and μ which are both the identity. This limitation and the stronger condition for compatibility recalled above make most storing problems impossible to solve.

Now we consider some diamond schemes.

Example 2.1 : Let us build the extension with the following parameters:

- $R = [0,3] \times [0,3]$
- $\phi_{|R} = i+4j$
- $\lambda(k) = k+4 \ \underline{\text{mod}}\ 16$
- $\mu(k) = \begin{cases} k+1 \ \underline{\text{mod}}\ 16 & \text{if } k \text{ is even} \\ k-1 \ \underline{\text{mod}}\ 16 & \text{if } k \text{ is odd} \end{cases}$ (exchange permutation)

λ and μ are commuting permutations of $\{0..15\}$. The extension is regular, as ϕ is a one-to one-mapping from R onto $\{0..15\}$. The result is depicted on Fig. 3, where the value inside the square (i,j) is the value of $\phi(i,j)$, i.e. the number of the memory bank where A_{ij} is stored.

Example 2.2 : The skewing schemes described by Van Voorhis [10] are diamond schemes extended from:

- $R_{N,1} = \{0..N-1\} \times \{0\}$ ($x = N$, $y = 1$)
- $\phi_{|R}$: any function with range $\{0..N-1\}$

by means of:

- $\lambda = Id$
- $\mu(k) = k+1 \ \underline{\text{mod}}\ N$.

Example 2.3 : Linear skewing schemes are a straightforward method for storing arrays in Fortran, Algol, Pascal and similar programming languages. Therefore, their properties have been investigated in depth both from a theoretical and from a practical point of view [3][4][5][7]. For two integers u and v, the linear skewing scheme $L_{u,v}$ is defined by $L_{u,v}(i,j) = ui+vj \ \underline{\text{mod}}\ N$.

A linear skewing scheme is a diamond scheme as stated below:

Proposition 2.1 : A linear skewing scheme $L_{u,v}$ is a diamond scheme with a reference rectangle reduced to the point (0,0) (i.e. $R_{1,1}$) and the following parameters :

- $\phi_{|R}(0,0) = 0$
- $\lambda(x) = x+u \quad \underline{\text{mod}}\ N$
- $\mu(x) = x+v \quad \underline{\text{mod}}\ N$

This is quite obvious, as by construction:

$$\lambda^i \mu^j ((0,0)) = ui+vj \quad \underline{\text{mod}}\ N$$

$R_{1,1}$ is not the only rectangle from which $L_{u,v}$ can be extended:

Proposition 2.2 : A linear skewing scheme is a regular diamond scheme iff gcd(u,v,N) = 1. As it can be easily verified, an adequate reference rectangle is $R_{x,y}$ with $x = N/\gcd(u,N)$ and $y = \gcd(u,N)$. Extension is made by means of $\lambda(x) = x+\text{lcm}(u,N)$ and $\mu(x) = x+\gcd(u,N)$.

It must be noticed that, if $\gcd(u,v,N) \neq 1$, then the skewing scheme is concentrated in a fraction of the banks, and thus of little practical interest. If N is a power of two, the condition is simply that one at least of u, v must be odd.

Example 2.4 : Rows, columns and diagonals.

Shapiro's results show that it is impossible to build a skewing scheme compatible with rows, columns and diagonals when the number of memory modules is a power of two, as it is impossible to tesselate the plane simultaneously with those templates. If the skewing scheme is a diamond scheme, one can easily prove that parallel access to all lines, all columns and the diagonal based at point (0,0) implies parallel access to all diagonals, and hence is impossible. We thus have to weaken our constraints; for example we may require parallel access to only a subset of columns.

For example, let us assume that parallel access is required to:

- all lines
- the columns shifted up by half height from each other
- the diagonal based at point (0,0)

for N a power of two.

Let us take for reference rectangle the column at point (0,0). Then, we will also have access to these templates shifted at any position along the x-axis, and the plane will be tesselated with rows, columns and diagonals accessible in parallel.

In order to have access to the row at point (0,0), λ must have a single cycle of length N; as a matter of fact, the value of ϕ at point (0,0) being equal to $\lambda^{i}(\phi(0,0))$, if λ has a cycle of length < N, ϕ will not be a bijection from the row located at point (0,0) onto {0..N-1}. We can assume $\lambda(i) = i+1$ mod N. Then, to have access to the diagonal, we must choose ϕ on the reference column such that $\phi(i)+i$ is one-to-one. $\phi(i) = 2i$ is a suitable choice. Then, by choosing $\mu(i) = i+1$ mod N, the column shifted up by half height is accessible in parallel. Permutations λ and μ quite obviously commute.

Thus, the diamond scheme based on :

- $R_{1,N}$
- $\lambda(i) = i+1$ mod N
- $\mu(i) = i+1$ mod N
- $\phi(i) = 2i$ mod N

answers our requirements. Fig. 4 shows the skewing scheme for N = 8.

Unscrambling requires only circular shifts. It must be noticed that reverse diagonals located at the same points as the diagonals can be accessed in parallel. This extension is <u>not</u> regular (whatever reference rectangle is used). This is a good example of a diamond scheme which is useful although non regular.

3. BASIC PROPERTIES OF EXTENSIONS, SEPARABLE EXTENSIONS.

We shall study here the basic properties of extensions : periodicity, effect of large translations , independance on $\phi_{|R}$ for regular extensions. The following lemma will be of use :

Lemma 3.1 : Let ϕ be an extension. For all translations π in Π, the following equality holds :

$$\phi \circ \pi = [\lambda,\mu]^{\pi} \circ \phi \qquad (3)$$

Proof : We will prove the equality over each rectangle of the reference covering. Let π' Π, we have, using (1) and (2) :

$$(\phi \circ \pi)_{|\pi'(R)} = \phi_{|\pi+\pi'(R)} \circ \pi_{|\pi'(R)}$$

$$= [\lambda,\mu]^{\pi+\pi'} \circ \phi_{|R} \circ (\pi\circ\pi')^{-1} \circ \pi$$

$$= [\lambda,\mu]^{\pi} \circ [\lambda,\mu]^{\pi'} \circ \phi_{|R} \circ \pi'^{-1}$$

$$= [\lambda,\mu]^{\pi} \circ \phi_{|\pi'(R)}$$

Thus (3) holds over Z^2 for all π in Π. □

This lemma is also useful to unscramble data for templates obtained by translation of a template by elements of Π.

Proposition 3.2 : Let r (resp. s) be the order of λ (resp. μ), i.e. the smallest positive integer such that $\lambda^r = Id$ (resp. $\mu^s = Id$). Then $\phi^{\lambda,\mu}$ is periodic, and its period is (rx, sy).

Proof : Apply lemma 3.1 with $\pi_{nr,ms}$ for (n, m) belonging to Z^2 :

$$\phi \circ \pi_{nr,ms} = [\lambda,\mu]^{\pi_{nr,ms}} \circ \phi$$

$$= (\lambda^r)^n \circ (\mu^s)^m \circ \phi = \phi$$

Thus, (rx,sy) is the period of ϕ. □

Proposition 3.3 : If a template T is compatible with ϕ, then, for any π Π, $\pi(T)$ is compatible with ϕ.

Proof : Apply lemma 3.1 to all points of T:

$$[\lambda,\mu]^{\pi} \circ \phi_{|T} = (\phi \circ \pi)_{|T}$$
$$= \phi_{|\pi(T)} \circ \pi_{|T}$$

As π and $[\lambda,\mu]^{\pi}$ are bijections, $\phi_{|\pi(T)}$ is a bijection. Then, $\pi(T)$ is compatible with ϕ. □

This proposition enables us to check the compatibility of a data template only on the translated template by Π that contains the origin (0,0). It will be the case in all our figures.

We shall now give, for regular extensions, an equivalent definition which will be more convenient for visualizing the effect of the permutations. For a regular extension ϕ, as $\phi_{|R}$ is a one-to-one mapping, we can introduce the conjugate of λ (resp. μ) with respect to $\phi_{|R}$:

$$\hat{\lambda} = \phi^{-1}_{|R} \circ \lambda \circ \phi_{|R}$$

$$\hat{\mu} = \phi^{-1}_{|R} \circ \mu \circ \phi_{|R}$$

$\hat{\lambda}$ and $\hat{\mu}$ are permutations of R, and they commute iff λ and μ commute.

Proposition 3.4 : ϕ is a regular extension of base R iff there exist two permutations $\hat{\lambda}$, $\hat{\mu}$ of R, commuting with each other, such that :

a) $\phi_{|R}$ is a one-to-one mapping onto $\{0..N-1\}$

b) for all π in Π, $\phi_{|\pi(R)} = \phi_{|R} \circ [\hat{\lambda},\hat{\mu}]^{\pi} \circ \pi^{-1}$

Proof : obvious. □

Using this alternate definition, we get the following property:

Proposition 3.5 : The set of templates compatible with a regular extension ϕ does not depend upon $\phi_{|R}$, but only upon $\hat{\lambda}$ and $\hat{\mu}$.

In other words, two regular extensions having the same R, $\hat{\lambda}$ and $\hat{\mu}$ have the same set of compatible templates.

Proof : Let ϕ, ϕ' be two regular extensions associated with the same permutations on R, $\hat{\lambda}$ and $\hat{\mu}$.

Let $\theta = \phi'_{|R} \circ (\phi_{|R})^{-1}$. θ is a permutation of $\{0..N-1\}$. The following equality holds:

$$\theta \circ \phi_{|R} = \phi' \circ \phi^{-1} \circ \phi_{|R} = \phi'_{|R}$$

Hence:

$$\begin{aligned}\theta \circ \phi_{|\pi(R)} &= \theta \circ \phi_{|R} \circ [\hat{\lambda},\hat{\mu}]^{\pi} \circ \pi^{-1} \\ &= \phi'_{|R} \circ [\hat{\lambda},\hat{\mu}]^{\pi} \circ \pi^{-1} \\ &= \phi'_{|\pi(R)}\end{aligned}$$

Thus, $\theta \circ \phi = \phi'$. If a template T is compatible with ϕ, it is also compatible with ϕ' because θ is a one-to-one mapping. □

This proposition will help to design the unscrambling algorithms by modifying the numbering on the reference template, i.e. $\phi_{|R}$, without changing any of the compatibility properties.

We will now define a special class of extensions obtained with a simple choice of $\hat{\lambda}$ and $\hat{\mu}$, which will simplify the computation of compatible templates.

Definition 3.1 : A separable extension is a regular extension where $\hat{\lambda}$ is a permutation of rows of R, $\hat{\mu}$ a permutation of columns of R.

In other words, there exists a permutation l (resp c) of {0..y-1} (resp. {0..x-1}) such that:

$$\hat{\lambda}(i,j) = (i,l(j)) \text{ (resp } \hat{\mu}(i,j) = (c(i),j)\text{)}$$

Then we have:

$$\phi(i,j) = \phi_{|R}(c^{j'}(i_0),\ l^{i'}(j_0))$$

where $i = xi' + i_0$ and $j = yj' + j_0$ are euclidian divisions.

Example 3.1 : Example 2.1, described in Fig.3, is in fact a separable extension. The row permutation l is a circular shift by one position, and the column permutation is a pairwise exchange of adjacent columns.

4. CYCLES AND ORBITS

In this section, E will denote a set of N elements, actually {0..N-1} or R for our purpose, λ and μ will be two commuting permutations over E.

We will examine some basic properties of permutations, that will enable us to split the sets {0..N-1} or R into subsets (cycles and orbits) on which the actions of λ and μ will be simpler to describe. This will be useful to build skewing schemes compatible with a given set of templates, by establishing constraints on cycles and orbits of λ, μ or $\hat{\lambda}$, $\hat{\mu}$.

First we will recall the definition and elementary properties of cycles.

Definition 4.1 : Let λ be a permutation on E. A cycle δ of λ is a minimal non-empty subset of E stable by λ:

$$\delta \text{ is a cycle of } \lambda \iff \begin{cases} \delta \neq \emptyset \\ \lambda(\delta) \subset \delta \\ \forall\, A \subset \delta,\ A \neq \emptyset:\ \lambda(A) \subset A \implies A = \delta \end{cases}$$

The number of elements of δ is the length of the cycle, $|\delta|$. Basic results concerning permutations are recalled in the following proposition a proof of which can be found in usual algebra textbooks.

Proposition 4.1 :

a) $\lambda^{|\delta|}$ is the identity over δ, and $|\delta|$ is the smallest integer k such that there exists an element z of δ verifying $\lambda^k(z) = z$.

b) δ is the disjoint union of the sets $\{\lambda^i(z)\}$ where i belongs to $\{0..|\delta|-1\}$ and z is a given point of δ.

c) The cycles of λ constitute a partition of E.

For example, if ϕ is a separable extension, cycles of $\hat{\lambda}$ are sets of columns of R and are obtained by translations of cycles of l, conjugated to R via ϕ.

Proposition 4.2 : Let λ, μ be two commuting permutations over E, let δ be a cycle of λ. Then, $\mu(\delta)$ is also a cycle of λ.

Proof :

1) $\mu(\delta)$ is stable by λ.

$$\lambda(\mu(\delta)) = \mu(\lambda(\delta)) = \mu(\delta) .$$

2) (minimality): Let $D \subset \mu(\delta)$, $D \neq \emptyset$, $\lambda(D) \subset D$. We will prove that $\mu^{-1}(D)$ is stable by λ:

$$\lambda(\mu^{-1}(D)) = \mu^{-1}(\lambda(D)) \subset \mu^{-1}(D).$$

As $\mu^{-1}(D)$ is not empty (D is not empty), $\mu^{-1}(D)$, stable by λ and included in δ, is equal to δ, a cycle. Thus $D = \mu(\delta)$ and $\mu(\delta)$ is a cycle. □

Definition 4.2 : Let λ, μ be two commuting permutations on E. An orbit Θ of λ and μ is a minimal non-empty subset of E stable by λ and μ.

$$\Theta \text{ is an orbit of } \lambda \text{ and } \mu \iff \left\{ \begin{array}{l} \Theta \neq \emptyset \\ \lambda(\Theta) \subset \Theta,\ \mu(\Theta) \subset \Theta \\ \left.\begin{array}{l} \forall A \subset \Theta,\ A \neq \emptyset \\ \lambda(A) \subset A,\ \mu(A) \subset A \end{array}\right\} \Rightarrow A = \Theta \end{array} \right.$$

The number of elements of Θ will be called the length of the orbit, noted $|\Theta|$. As we shall see, orbits have properties very similar to cycles.

Example 4.1 : Orbits of separable extensions. The orbits of a separable extension using l and c as row and column permutations are the cartesian products of cycles of l and c, when interpreted geometrically. Fig. 5.a shows the cycles of the l and c permutation of the separable extension described in example 3.1 (Fig. 3). Fig. 5.b shows the corresponding two orbits, obtained as cartesian products of the cycles.

Example 4.2 : Let us consider a sligthly more complex example. Let

- $E = \{0..7\}$
- λ, μ defined by the table:

i	$\lambda(i)$	$\mu(i)$
0	1	3
1	0	2
2	3	0
3	2	1
4	5	6
5	4	7
6	7	4
7	6	5

The reader will check that λ and μ do commute. The cycles of λ are (0,1) (2,3) (4,5) (6,7), the cycles of μ are (0,3,1,2), (4,6), (5,7). There are two orbits, $\Theta_1 = \{0,1,2,3\}$ and $\Theta_2 = \{4,5,6,7\}$.

Proposition 4.3 :

a) An orbit may be partitionned into cycles of λ of equal length and into cycles of μ of equal length.

b) Let θ be an orbit for λ and μ, let z be a point in θ. Then, there exist two integers r and s, such that θ is the disjoint union of the sets $\{\lambda^i o\ \mu^j(z)\}$ for (i,j) belonging to $[0..r-1]\times[0..s-1]$.

c) Orbits of λ and μ form a partition of E.

Proof : This proposition stems directly from propositions 4.1 and 4.2. □

Note that statement b) is a two-dimensional extension of the similar statement on cycles.

Representation of cycles and orbits.

Proposition 4.1 enables us to identify a cycle δ of λ with the group of integers modulo the length $|\delta|$ of the cycle, by the application:

$$
\begin{aligned}
Z/|\delta|Z &\longrightarrow \delta \\
i &\longrightarrow \lambda^i(z)
\end{aligned}
$$

where z is an arbitrary element of δ.

The action of λ on δ is then equivalent to an addition by one in this group. The equivalent identification for orbits is unfortunately not to the rectangle suggested by the proposition 4.3, but to a more general structure.

Proposition 4.4 : Let λ and μ be two commuting permutations on E, θ an orbit of λ and μ, z an arbitrary point of θ. Then the set of integers (m,n) such that $\lambda^m \circ \mu^n$ is the identity over θ is a lattice L (i.e. a two-dimensional subgroup) of Z^2. The orbit θ may be identified with the quotient group Z^2/L by the mapping:

$$\psi : \quad Z^2/L \longrightarrow \theta$$

$$(i,j) \longrightarrow \lambda^i \circ \mu^j(z)$$

The action of λ (resp. μ) in θ is thus identified to an x-wise (resp. an y-wise) translation in Z^2 modulo the lattice L.

The inverse mapping from θ onto Z^2/L will be noted as $a \longrightarrow \bar{a}$ where a belongs to θ.

Proof : obvious. □

The quotient group Z^2/L may be geometrically interpreted as a parallelogram folded onto a torus. The sides of the parallelogram form a basis of the lattice L. A lattice on Z^2 does not have a unique basis, thus L may be represented by several parallelograms, each of them associated to a different basis of L. Two bases are of particular interest; the first one has a base vector U_x parallel to the x axis: $U_x = (u_x,0)$, $u_x > 0$; the second has a base vector V_y parallel to the y axis: $V_y = (0,v_y)$, $v_y > 0$. Such bases always exist. The length of U_x is equal to the length of the cycles of λ in the orbit θ, and the length of V_y is equal to the length of the cycles of μ in θ.

Example 4.3 : The lattices associated with separable extensions are always rectangular, i.e. have an orthogonal basis, with base vectors parallel to the x-axis (resp. y-axis). If δ, δ' are cycles of the 1 and c permutations, the cartesian product $\delta\times\delta'$ is an orbit and its lattice is obtained as:

$$L = |\delta| Z \times |\delta'| Z$$

Example 4.4 : Using proposition 4.4, let us examine the structure of the orbits for λ and μ as described in example 4.2. The lattice L_1 associated to the first orbit Θ_1 is generated by the two vectors $U_1 = (2,0)$, $V_1 = (1,2)$. The lattice L_2 associated with the second orbit is generated by $U_2 = (0,2)$, $V_2 = (2,0)$. These lattices are represented in figure 6.

The structure of L_2 is described by a single rectangle. L_1 examplifies the more general case. Orbit Θ_1 consists of two cycles of λ, each having two elements, but only of a single cycle of μ having four elements. Fig. 7 describes the action of λ and μ on the representation of Θ_1.

The main interest of proposition 4.4 is that it will allow us to transport equations on orbits onto the group Z^2/L, i.e. a parallelogram, where they will be easier to solve.

Conversely, given k numbers $n_1,\ldots,n_k$ such that $n_1+n_2+\ldots+n_k = N$, it is possible to build a permutation the cycles of which have precisely the lengths $n_1,\ldots,n_k$. Similarly, given k lattices of Z^2, $L_1,\ldots,L_k$, such that the sum of their areas (i.e. the area of the associated parallelograms) is equal to N, it is possible to build two permutations λ and μ commuting with each other and such that their orbits have exactly the structures described by $L_1,\ldots,L_k$.

5 Compatible templates

We shall first establish a general theorem expressing the conditions on the parameters λ, μ, $\phi_{|R}$ of a diamond scheme ϕ, for a given template to be compatible with ϕ. Then we shall study the two cases where the template has a 'regular' geometric structure and where the extension is separable. All templates considered in this section will have exactly N points, N being the number of memory banks. Thus T will be compatible with ϕ iff $\phi_{|T}$ is bijective.

Definition 5.1 : Let T be a template; the decomposition of T on the reference covering is a set of distinct points $\{a_i\}$ in R and, for each point a_i, a set of translations $C(a_i)$ in Π. These sets have the property that:

$$T = \bar{\bigcup_{a_i}} \left\{ \bar{\bigcup_{\pi \in C(a_i)}} \pi(a_i) \right\}$$

$\bar{\bigcup}$ is used to denote the union of disjoint sets.

The decomposition is obviously unique. The points a_i are the elements of R such that some $\pi(a_i)$ lies in T, and the set $C(a_i)$ is made of all translations having this property. Fig. 8 gives an example of decomposition.

Each element π of $C(a_i)$ being a translation by a vector of the form (ux,vy), we denote by $M(a_i)$ the set of associated points (u,v); that is:

$$M(a_i) = \{ (u,v) \in Z^2 / (ux,vy) \in C(a_i) \}$$

Lemma 5.2 : Let T be a template, λ, μ, $\phi_{|R}$ an extension, $\{a_i, C(a_i)\}$ the decomposition of T. T is compatible with ϕ iff, for all orbits Θ of λ and μ, the following equality holds:

$$\Theta = \bigcup_{\substack{a_i \text{ such that} \\ \phi(a_i) \in \Theta}} \left\{ \bigcup_{\pi \in C(a_i)} [\lambda,\mu]^{\pi} \phi(a_i) \right\} \tag{4}$$

Proof : Whether T is compatible with ϕ or not, by using the decomposition of T, the subset $\phi(T)$ verifies:

$$\phi(T) = \phi \left\{ \bar{\bigcup_{a_i}} \left\{ \bar{\bigcup_{\pi \in C(a_i)}} \pi(a_i) \right\} \right\} \quad \text{(disjoint unions)}$$

$$= \bigcup_{a_i} \left\{ \bigcup_{\pi \in C(a_i)} \phi o \pi(a_i) \right\} \quad \text{(\underline{not} necessarily disjoint unions!)}$$

$$= \bigcup_{a_i} \left\{ \bigcup_{\pi \in C(a_i)} [\lambda,\mu]^{\pi} \phi(a_i) \right\} \tag{5}$$

T is compatible with ϕ iff $\phi(T)$ has the same cardinality as T, i.e. N, that is iff the sets whose union is considered in equation (5) above are disjoint.

a) if T is compatible with ϕ, let Θ be an orbit of λ and μ. Then, Θ is equal to $\phi(T) \cap \Theta$ since $\phi(T) = \{0..N-1\}$. Then, using (5), we find that:

$$\Theta = \Theta \cap \phi(T)$$

$$= \bar{\bigcup_{\substack{a_i \text{ such that} \\ \phi(a_i) \in \Theta}}} \left\{ \bar{\bigcup_{\pi \in C(a_i)}} [\lambda,\mu]^{\pi} \phi(a_i) \right\}$$

because the integers $[\lambda,\mu]^{\pi} \phi(a_i)$ are in Θ if $\phi(a_i)$ belongs to Θ and only in this case (stability of an orbit by λ and μ).

b) Conversely, let us assume that (4) holds for all orbits. From equation (5), it follows at once that all orbits are contained in $\phi(T)$. Therefore, $\phi(T)$ is equal to their union, $\{0..N-1\}$, which means that T is compatible with ϕ. □

In section 4, we have associated with each orbit Θ a lattice L such that Θ is mapped onto the group Z^2/L by a canonical isomorphism $x \longrightarrow \bar{x}$. From this identification, we get another statement of lemma 5.2, which gives rise to easier computation and suggests algorithms for testing compatibility.

<u>Theorem 5.3</u> : T is compatible with ϕ iff, for each orbit Θ of λ and μ, the group Z^2/L associated with Θ is obtained by:

$$Z^2/L = \bar{\bigcup_{\phi(a_i) \in \Theta}} \left\{ \overline{\phi(a_i)} + M(a_i) \right\} \qquad (6)$$

where $x \longrightarrow \bar{x}$ is the natural mapping from Θ onto Z^2/L and where the $M(a_i)$ are mapped from Z^2 into Z^2/L.

This theorem has thus established an equivalence between the problem of compatibility and the problem of partitionning a set of parallelograms, the Z^2/L groups. We will now examine a case where this partition is simple.

<u>Definition 5.2</u> : A template T is uniform iff all the $M(a_i)$ of its decomposition on R are identical to the same rectangle M in Z^2, of the form:

$$M = \{ (h,k) \in Z^2 \ / \ 0 \leq h < H, \ 0 \leq k < K \} \qquad H, K \geq 1$$

The template is said to be one-dimensional if M is an horizontal or vertical segment, two-dimensional otherwise. The set of base points of T, $\{a_i\}$, is called the atom of T, and will be noted A. It is noteworthy that HKcard(A) = N, since T has N elements.

Fig. 9 gives examples of regular templates, with reference $R_{4,4}$.

Theorem 5.4 : Let T be a uniform template, ϕ a diamond scheme. T is compatible with ϕ iff, for all orbits Θ of λ and μ, H divides the length of the cycles of λ in Θ, K divides the length of the cycles of μ in Θ, HK card($\phi(A) \cap \Theta$) = card(Θ) and one of the two conditions holds:

a) The intersection of $\phi(A)$ and any cycle of λ in Θ is either the empty set or a cycle of λ^H, and the intersection of $\phi(A)$ and Θ is contained in an orbit of λ and μ^K.

b) The intersection of $\phi(A)$ and any cycle of μ in Θ is either the empty set or a cycle of μ^K, and the intersection of $\phi(A)$ and Θ is contained in an orbit of λ^H and μ.

Proof : For a uniform template, (6) in theorem 5.3 reduces to:

$$Z^2/L = \overline{\bigcup_{v \in V}} \ (\bar{v} + M)$$

where $V = \phi(A) \cap \Theta$ and M is the rectangle $0 \leq h < H$, $0 \leq k < K$. This equation expresses that Z^2/L is paved by rectangles based at the points v.

Let us assume that T is compatible with ϕ. Then, for a given orbit, we have a paving of Z^2/L by rectangles, which naturally can be extented to a tesselation of Z^2 by the same rectangles. This tesselation may be achieved only in two ways: the rectangles must be drawn up horizontally or vertically as shown on Fig.10. Let us assume that the first is true. Then, the base points of the tesselation of Z^2 lies on some horizontal lines spaced by steps of K and along these horizontal are regularly spaced by steps of H.

If we return to the interpretation in the orbit, this means that the intersection of V and a cycle of λ (i.e. an horizontal segment in Z^2) is empty or a cycle of λ^H (spaced by H) and that V is included in an orbit of λ and μ^K , as the horizontal lines containing the points of V are spaced by K. These are the conditions enumerated in a). If we had assumed a vertical tesselation, we would have obtained the set of conditions b).

These conditions are not sufficient, as they only used the fact that Z^2 is tesselated by the rectangle M. We will now use the fact that this tesselation is also valid for Z^2/L.

Let us consider a basis of the lattice having an horizontal vector. The length of this vector will be exactly the length of the cycles of λ in Θ. H must then divide this length, so that the tesselation of Z^2 may be folded into a paving of Z^2/L. A similar argument shows that K must divide the length of the cycles of μ in Θ. Finally, an obvious counting argument shows that HKcard(V) = card(Θ).

Conversely, conditions a) or b) and the fact that HKcard(V) = card(Θ) establish a tesselation of Z^2. The divisibility conditions on H and K show that this tesselation may be folded into a paving of Z^2/L by rectangles, which proves the result. □

The practical interest of this theorem lies in the following two corollaries.

<u>Proposition 5.5</u> : Let T be a one-dimensionnal uniform template with K = 1 (resp. H = 1), ϕ a diamond scheme. T is compatible with ϕ iff H

(resp. K) divides the length of the cycles of λ (resp. μ), and the intersection of $\phi(A)$ with a cycle of λ (resp. μ) is a cycle of λ^H (resp. μ^K).

<u>Proof</u> : Let us assume that T is one-dimensional with K = 1. M is then reduced to an horizontal segment (i,0) $0 \leq i < H$.

If T is compatible with ϕ, the tesselation induced on Z^2 will necessarily be of the horizontal kind, as, in this case, vertical tesselations are also horizontal tesselations. Then the points of V along a horizontal line are spaced by H, and V has points on all horizontal lines. In terms of cycles, this means that the intersection of $\phi(A)$ and a cycle of λ (i.e. a horizontal line) is exactly a cycle of λ^H. As in the main theorem, H must divide the length of the cycles of λ. The counting argument is unnecessary, as there are points of V on each horizontal line.

Conversely, the hypothesis on cycles of λ gives a tesselation of Z^2 by horizontal segments, which may be transformed into a paving of Z^2/L because H divides the length of the cycles. □

<u>Proposition 5.6</u> : Let T be a uniform template, for which the atom A is a cartesian product (i.e. $A = A_x \times A_y$), let ϕ be a separable extension with parameters l and c. T is compatible with ϕ, iff for all cycles δ of l (resp. c), the intersection of δ with A_x (resp. A_y) is a cycle of l^H (resp. c^K) (thus, non-empty).

<u>Proof</u> : The orbits of a separable extension are cartesian products of cycles of l and c. Thence, conditions a) and b) of theorem 5.4 and the fact that A is a cartesian product imply that the base points of the

tesselation (i.e. the intersection of A with an orbit) lie regularly on the grid $H\mathbb{Z}\times K\mathbb{Z}$ (i.e. an orbit of $\hat{\lambda}^H$ and $\hat{\mu}^K$); so, the intersection of A with any orbit of $\hat{\lambda}$ and $\hat{\mu}$ must be an orbit of $\hat{\lambda}^H$ and $\hat{\mu}^K$ (i.e. a cartesian product of a cycle of l^H and a cycle of c^K). A counting argument proves that if this is possible, then H and K divide the length of the cycles of $\hat{\lambda}$ and $\hat{\mu}$ in $\hat{\theta}$, and that HK divides the cardinal of $\hat{\theta}$. the converse proof is similar. □

6. APPLICATIONS.

In this section, we will find skewing schemes suitable for certain families of templates. For each case, we will deduce from the theorems developed in the previous sections constraints on the permutations λ and μ, and then exhibit "simple" permutations meeting these constraints. An easy unscrambling is an important argument in the choice of λ and μ.

1) Access to an arbitrary template.

Using the ideas previously developped, we will now show that, given a reference rectangle R and an arbitrary template T such that T is covered by four copies of R in a square pattern as in the example of Fig.8, it is possible to build a regular diamond scheme with which T and R are both compatible. Thus, it is possible to access in parallel an (almost) arbitrary template and a rectangle. Throughout the proof of ths assertion we will refer to Fig.8 as an example.

Up to a translation, we can assume that the four copies of R covering T are obtained by the translations (0,0), (x,0), (0,y), (x,y), associated with the permutations Id, λ, μ, $\lambda o \mu$. We will build a regular diamond scheme based on R such that each orbit of λ and μ contains exactly one integer $\phi(x)$ where x is in the decomposition of T on R. Let us choose for ϕ an arbitrary bijection on R. On Fig.8, we will choose $\phi(i,j) = i + 4j \ \underline{\text{mod}}\ 8$, for (i,j) belonging to $R_{4,2}$.

The decomposition of T may be written as:

$$T = \bar{\bigcup_{a_i}} \left\{ \bar{\bigcup_{\pi \in C(a_i)}} \pi(a_i) \right\}$$

In our example, T is decomposed with the following values for a_i and $C(a_i)$:

$$a_1 = (3,1) \qquad C(a_1) = \{(1,0)\}$$
$$a_2 = (0,1) \qquad C(a_2) = \{(1,1)\}$$
$$a_3 = (1,0) \qquad C(a_3) = \{(1,1)\}$$
$$a_4 = (2,1) \qquad C(a_4) = \{(1,0),\ (1,1)\}$$
$$a_5 = (1,1) \qquad C(a_5) = \{(1,0),\ (1,1),\ (0,1)\}$$

In the general case, there are 15 possible sets $C(a_i)$. For each set, we will build an orbit Θ_{a_i} containing one value $\phi(a_i)$ and we will define over Θ_{a_i} two permutations, λ and μ, commuting with each other and verifying equation (4) in lemma 5.2. Thus, we will have built λ and μ on $\{0..N-1\}$ that are solutions of the problem.

The orbit Θ_{a_i} associated with a_i will contain exactly $card(C(a_i))$ integers, one of them being $\phi(a_i)$ and the others chosen in $\{0..N-1\}$, such that the sets Θ_{a_i} form a partition of $\{0..N-1\}$. An obvious counting argument that such a choice is always possible. In our example, we can choose the Θ_{a_i} as follows:

- $\phi(a_1) = 7$ $\qquad card(\Theta_{a_1}) = card(C(a_1)) = 1$
  $\Theta_{a_1} = \{7\}$
- $\phi(a_2) = 4$ $\qquad card(\Theta_{a_2}) = card(C(a_2)) = 1$
  $\Theta_{a_2} = \{4\}$
- $\phi(a_3) = 1$ $\qquad card(\Theta_{a_3}) = card(C(a_3)) = 1$
  $\Theta_{a_3} = \{1\}$
- $\phi(a_4) = 6$ $\qquad card(\Theta_{a_4}) = card(C(a_4)) = 2$
  $\Theta_{a_4} = \{6,0\}$
- $\phi(a_5) = 5$ $\qquad card(\Theta_{a_5}) = card(C(a_5)) = 3$
  $\Theta_{a_5} = \{5,2,3\}$

These orbits respect the aforementionned conditions.

Now, we will define λ and μ on each Θ_{a_i} such that Θ_{a_i} will be composed exactly of the integers $[\lambda,\mu]^{\pi}$ where π belongs to $C(a_i)$. Four major cases must be examined, according to the cardinality of $C(a_i)$.

1) $card(C(a_i)) = 1$. The orbit is reduced to the set $\{\phi(a_i)\}$, and we define λ and μ over it as the identity:

$$\lambda(\phi(a_i)) = \phi(a_i) \qquad \mu(\phi(a_i)) = \phi(a_i)$$

In our example, a_1 , a_2 , and a_3 have single-point orbits. Thus, λ and μ are the identity on Θ_{a_1} , Θ_{a_2} and Θ_{a_3} :

$$\lambda(7) = 7 \qquad \mu(7) = 7$$

$$\lambda(4) = 4 \qquad \mu(4) = 4$$

$$\lambda(1) = 1 \qquad \mu(1) = 1$$

2) $card(C(a_i)) = 2$. There are six cases. Let us consider the case with $C(a_i) = \{(1,0),(1,1)\}$. Then, if the orbit is $\{y_1,y_2\}$ with $y_1 = \phi(a_i)$, if we define λ and μ by:

$$\lambda(y_1) = y_2 \qquad \mu(y_1) = y_1$$

$$\lambda(y_2) = y_1 \qquad \mu(y_2) = y_2$$

Permutation μ (resp. λ) is the identity (resp. exchange permutation). Then λ and μ obviously commute and (4) is verified. The five other cases have similar solutions.

In our example, Θ_{a_4} has two points, and $C(a_4)$ has exactly the configuration considered above. Hence, λ and μ are defined on Θ_{a_4} as:

$$\lambda(6) = 0 \qquad \mu(6) = 6$$

$$\lambda(0) = 6 \qquad \mu(0) = 0$$

3) $card(C(a_i)) = 3$. There are four cases. Let us consider the case with $C(a_i) = \{(1,0),(0,1),(1,1)\}$. Then, if the orbit is $\{y_1,y_2,y_3\}$ with $\phi(a_i) = y_1$, if we define λ and μ by:

$$\lambda(y_1) = y_3 \qquad \mu(y_1) = y_2$$
$$\lambda(y_2) = y_1 \qquad \mu(y_2) = y_3$$
$$\lambda(y_3) = y_2 \qquad \mu(y_3) = y_1$$

then, λ and μ commute (in fact $\lambda = \mu$) and (4) is verified. The five other cases have similar solutions.

In our example, θ_{a_5} has three points and $C(a_5)$ has exactly the configuration considered above. Hence, λ and μ are defined on θ_{a_5} as:

$$\lambda(5) = 3 \qquad \mu(5) = 2$$
$$\lambda(2) = 5 \qquad \mu(2) = 3$$
$$\lambda(3) = 2 \qquad \mu(3) = 5$$

4) $\mathrm{card}(C(a_i)) = 4$. The orbit is $\{y_1,y_2,y_3,y_4\}$. If we define λ and μ by:

$$\lambda(y_1) = y_2 \qquad \mu(y_1) = y_3$$
$$\lambda(y_2) = y_3 \qquad \mu(y_2) = y_4$$
$$\lambda(y_3) = y_4 \qquad \mu(y_3) = y_1$$
$$\lambda(y_4) = y_1 \qquad \mu(y_4) = y_2$$

then, λ and μ commute and (4) is verified.

In our example, we have no orbit with four elements.

Thus, we have built permutations λ and μ on $\{0..N-1\}$, having the required orbits such that, on each orbit, the conditions of lemma 5.2 are verified. Then, T is compatible with the regular extension from ϕ by λ and μ, which solves our problem. In our example, λ and μ have been built on each orbit and the resulting scheme is depicted on Fig.11.

This method appears to be quite clumsy, as it consists in examining a rather large number of cases. Actually, a simple explanation shows how and why it works. The problem lies in proving that adequate permutations exist on each of our orbits.

If $C(a_i)$, considered as a subset of Z^2, tesselates Z^2 by translations forming a lattice L, then $C(a_i)$ may be identified to Z^2/L and hence to the orbit itself. As it is always possible to build two permutations having an orbit associated with a given lattice, thus we have built permutations λ and μ on our orbit that verify (4).

In our case, up to symetries, there are five cases, described on Fig.12, for $C(a_i)$, and each of them tesselates regularly the plane. Hence, a solution exists.

It must be noticed that this proof does not work if the covering of T by R is different, for example obtained by translations (0,0),(0,1),(0,2) and (0,3).

2) An organization for image processing.

On a 2^p memory bank system, we want to have parallel access to rectangles having different shape factors: $2^n \times 2^{p-n}$. Using Shapiro's theory, it is easy to show that it is impossible to access in parallel all those rectangles located at any position on the plane if 3 or more rectangular shapes are required. Instead, we will assume that parallel access to rectangles of a given shape is required only for a subset that covers the plane. This choice of templates is useful for image processing and generation as it permits parallel access to shapes with various form factors. For example, such an organization would allow to draw vectors in parallel in a minimum number of memory access.

We will choose as reference rectangle the rectangle $R_{1,2^p}$ ($X = 1$, $Y = 2^p$), i.e. a segment of vertical line. This choice is natural, since by Proposition 3.3, it will ensure that, if a template is compatible, all templates obtained by an x-wise translation will also be compatible. We will require parallel access to all rectangles of size $2^n \times 2^{p-n}$ located at points $(h2^n, k2^{p-n})$; for each form factor, the rectangles selected above tesselate the plane. Fig.13 shows the templates to which parallel access is required for p = 3. All the required templates are regular and one-dimensional.

Let $E_{n,k}$ be the vertical segment of length 2^{p-n} located at $(0,k2^{p-n})$, for k in $\{0..2^{n-p}-1\}$. For a given n, the $E_{n,k}$ form a partition of our base rectangle, which is equal to $E_{0,0}$. In fact, $E_{n,k}$ is obtained by the union of $E_{n+1,2k}$ and $E_{n+1,2k+1}$.

The rectangle of size $2^n \times 2^{p-n}$ located at $(0,k2^{p-n})$ is a one-dimensional template of atom $E_{n,k}$. If we apply Proposition 5.5, we see that, if δ is a cycle of λ, then for any $E_{n,k}$, $\phi(E_{n,k}) \cap \delta$ is a cycle of λ^{2^n} . Considering $E_{p,0}$, which is reduced to the single point (0,0), we see that all cycles of λ contain the point $\phi(0,0)$, hence λ has a single cycle of size 2^p. Hence, any $\phi(E_{n,k})$ is a cycle of λ^{2^n}.

As we assumed that ϕ was regular, this means that λ has a single cycle of length 2^p, which splits into 2 cycles of λ^2 of length 2^{p-1}, and so on. The simple permutation λ: $i \longrightarrow i+1$ has cycles with this property. The remaining problem is to build a skewing scheme on $E_{0,0}$ (i.e. $R_{1,2^p}$) that maps $E_{n,k}$ into a cycle of λ^{2^n} : $i \longrightarrow i+2^n$.

If we examine the binary decomposition of the elements of an $E_{n,k}$ and of the elements of a cycle of λ^{2^n}, we have:

$$E_{n,k} \;:\; j_p j_{p-1} \cdots j_{p-n+1} \underbrace{XXX\ldots XX}_{n \text{ bits}}$$

$$\delta(\lambda^{2^n}) :\; \underbrace{YYY\ldots YY}_{n \text{ bits}} i_p i_{p-1} \cdots i_{p-n+1}$$

It is thus natural to find that the bit-reversal permutation (noted BR) does map the $E_{n,k}$ into cycles of λ^{2^n}.

Then, a solution to the problem is to use:

- $R = R_{1,2^p}$
- $\mu = Id$
- $\lambda: i \longrightarrow i+1 \quad \underline{\text{mod}}\ 2^p$
- $\phi(0,j) = BR(j)$ on $R_{1,2^p}$.

Fig.14 gives the full skewing scheme for $p = 3$. The function ϕ is easily computed on the entire plane by:

$$\phi(i,j) = i + BR(j\ \underline{\text{mod}} 2^p) \qquad \underline{\text{mod}}\ 2^p.$$

The choice of λ simplifies greatly the unscrambling of data. The reader may check that, in order to map the rectangle of size $2^n \times 2^{p-n}$ located at point $(i,j2^{p-n})$ onto the rectangle of size $2^n \times 2^{p-n}$ located at point (0,0), the unscrambling permutation required is the circular shift:

$$x \longrightarrow x - (i + BR(j2^{p-n}\ \underline{\text{mod}}\ 2^p)) .$$

In fact, choosing $\mu = Id$ has for consequence that columns are accessible in any position, as well as rows. Unscrambling of rows requires a simple circular shift. Columns in a random position are more complex to unscramble, but the permutation may be performed by an Ω network, as it can be expressed by a circular shift (x-wise displacement) followed by a permutation of the form $x \longrightarrow BR(y+BR(x))$, which is Ω-pass as shown by Feng [13] and Steinberg [12] .

3) Multigrid methods and Total Reduction (T.R.) Fast solvers.

For this class of numerical methods [14] [15] [16], it is necessary to have parallel access to blocks and all levels of blocks on coarser grids; i.e. accesses by steps of $1,2,4,\dots,2^k$. Fig.15 shows an example of such data accesses. We will suppose that the number of memory banks is an even power of two: $N = 2^{2p}$. It must be noticed that under Shapiro's conditions, it is impossible to access efficiently more than one grid level. The advantage of our skewing scheme is that it allows efficient parallel access on different grid levels, this without any data reorganization even when handling various geometries or using local refinement techniques [17] [18].

In terms of extensions, it is natural to take as reference rectangle the block $R_{2^p,2^p}$. Then all the required templates are regular, and their atoms are cartesian products. We shall thus look for a separable extension, in order to use Proposition 5.6. If we consider the block with a step of 2^k, this proposition gives as necessary and sufficient condition that, if A is the atom, θ an orbit of λ and μ, $\phi(A) \cap \theta$ is an orbit of λ^{2^k} and μ^{2^k}. For k = n, that is for the coarsest possible grid, A is reduced to a single point, thus λ^{2^p} and μ^{2^p} are the identity function. If we interpret the conditions in terms of the row and column permutations, l and c, we find that the cycles of l^{2^k} and c^{2^k} are made up of integers regularly stepped by 2^k. An obvious solution is to use for l and c a circular shift of one position:

$$l(i) = c(i) = i+1 \qquad \underline{\text{mod}}\ 2^p$$

Thus, if we take $R = R_{2^p,2^p}$, ϕ being the natural numbering of R (i.e. $\phi_{|R}(i,j) = 2^p i+j$), $\hat{\lambda}$ the circular shift by one position of rows and $\hat{\mu}$ the circular shift by one position of columns, we have parallel access to $2^p \times 2^p$ blocks, and coarser blocks of the form:

$$(x_0,y_0) + (a.2^k,b.2^k) \qquad 0 \le a < 2^p \quad 0 \le b < 2^p$$
$$0 \le x_0 < 2^k \quad 0 \le y_0 < 2^k$$

Fig.13 gives the full skewing scheme for p = 2.

This skewing scheme also gives parallel access to all rows and columns, as is easily checked by application of Proposition 5.6.

Computation of $\phi(i,j)$ is quite simple. Let $i = (i_2, i_1, i_0)$, $j = (j_2, j_1, j_0)$ be the binary decomposition of i and j where i_1, j_1, i_0, j_0 are p bits wide. Then, we have:

$$\phi(i,j) = (j_1 + i_0 \, , \, i_1 + j_0)$$

The reader should notice that all coarse blocks are not available, only enough for a covering of Z^2. For example, the block by steps of two located at point (3,3) in Fig.16 is not accessible in parallel. All the unscrambling permutations induced by the computations on a grid level can be achieved by a single passage through an Ω-network, only when changing grid levels (fine to coarse or coarse to fine), we need two passes through an Ω-network or a Benes network. It must be noticed that the command of all the permutations required can be easily realized by the same techniques as these developped by Lenfant [20].

7. CONCLUSION AND GENERALIZATION.

The diamond schemes introduced in this paper are skewing schemes derived from a rectangular tesselation of the integral plane by means of two permutations λ and μ commuting with each other. Let us consider a rectangle T of this reference covering obtained from the reference rectangle by a displacement of K rectangles along the x-axis and L rectangles along the y-axis. Each element within T is stored in a memory bank number of which is the image by $\lambda^K \circ \mu^L$ of the memory bank number in which lies the corresponding element of the reference rectangle.

We have linked the compatibility between a diamond scheme and an access template to the existence of a tesselation of Z^2/L where the L's are integral lattices reflecting the structure of the orbits of permutations λ and μ. A general result (theorem 5.3) has been adapted to the specific case of uniform templates (theorem 5.4).

Various examples presented in this paper show that diamond schemes are a valuable tool to build skewing schemes adequate for a large number of realistic programs. The problem of storing several nested arrays as a single array to implement a multigrid algorithm shows the flexibility of our concept of compatibility. All grid levels can be handled efficiently. On each grid the programmer enjoys parallel access to a family of blocks covering the grid and to blocks neighbouring them so that the desired computations (e.g. interpolations) can be performed without memory conflicts. It is noteworthy that many blocks have distinct elements in the same memory bank. The requirement that all blocks can be accessed in parallel is too strong and can be satisfied at only one grid level.

The definition of a diamond scheme from its values on a reference pattern by means of a tesselation of the plane is quite similar to the mechanism used by Shapiro. In this pioneer work, λ and μ are both the identity but the tesselation is less restricted than ours; the reference pattern can be any N-element subset of the plane and it is replicated by a set of translations to cover the plane. The reader can check that most of our results still hold if the reference pattern is not a rectangle. However it is essential to our analysis that the underlying set of permutations be a subgroup of $\mathbf{Z}^2$. Although not mandatory in Shapiro' work, this assumption happens to be valid in his examples.

Our definitions and results may be readily generalized to n-dimensional arrays, using a set of 2n permutations, $(\lambda_1,\ldots,\lambda_n)$ and $(\mu_1,\ldots,\mu_n)$, commuting with each other (i.e. $\lambda_i \circ \mu_j = \mu_j \circ \lambda_i$, $\mu_i \circ \mu_j = \mu_j \circ \mu_i$ and $\lambda_i \circ \mu_j = \mu_j \circ \lambda_i$). A deeper generalization is obtained by noticing that the properties of extensions depend heavily upon the fact that the application from Π into S_N , defined by:

$$(\Pi,+) \longrightarrow (S_N,\circ)$$

$$\pi \longrightarrow [\lambda,\mu]^{\pi}$$

is a group homomorphism. By modifying the group structure on Π and choosing other homomorphisms from Π to S_N , an additional degree of freedom is gained. In all cases, the basic properties of orbits described in section 4 remain valid, and most of the results in section 5 still hold provided that the right group structures are used. Geometrical interpretation becomes more difficult, due to the n-dimensional structure to the problem and to the non-standard groups used. A full presentation of these generalizations would have obscured the paper, as most of them gain only very little in terms of the class of templates accessible.

There is one of these generalizations that is of practical interest, for very regular templates. In examples 2 and 3 of section 6, the skewing schemes where obtained over $\mathbf{Z}^2$ as:

$$\phi_2(i,j) = i + BR(j \ \underline{\text{mod}}\ 2^p) \qquad \underline{\text{mod}}\ 2^p$$

$$\phi_3(i,j) = (j_1 + i_0 , j_0 + i_1)$$

Here we have considered the pairs of integers less than 2^p as indices in a 2-dimension array. Alternatively we can consider them as sequences of 2^{p+1} bits and interpret them as elements of the vector space with dimension p+1 over Z/2Z (only the structure of additive groups matters). Using the generalized theorems and returning to the 2-dimension array we can prove easily that the skewing schemes defined over Z^2 by:

$$\phi'_2 (i,j) = i \oplus BR(j \ \underline{\text{mod}}\ 2^p) \qquad \underline{\text{mod}}\ 2^p$$

$$\phi'_3 (i,j) = (j_1 \oplus i_0 , j_0 \oplus i_1)$$

(where $\oplus$ is the Bitwise Exclusive Or) allow parallel access to the same templates as ϕ_1 and ϕ_2. Replacing additions by bitwise boolean operations is very iteresting for it suggests a simple and elegant solution for hardware implementation. On the same way the linear skewing scheme:

$$\phi(i,j) = i+j \qquad \underline{\text{mod}}\ 2^p$$

can be replaced by the scheme built into STARAN processor [19]:

$$\phi(i,j) = i \oplus j \qquad \underline{\text{mod}}\ 2^p$$

BIBLIOGRAPHY

[1] V.E.Benes, Mathematical Theory of Connecting Networks and Telephone Traffic. New York: Academic, 1965.

[2] D.H.Lawrie, "Memory-processor connection networks," University Illinois, Urbana-Champaign, Dep. Comput. Sci., Rep. 557, Feb. 1973.

[3] H.D.Shapiro, "Theoretical limitations on the use of parallel memories," Ph.D. Thesis, University Illinois, 1976.

[4] H.D.Shapiro, "Theoretical limitations on the efficient use of parallel memories," IEEE Transactions on Computers, Vol. C-27, pp. 421-428, May 1978.

[5] P.Budnick, D.J.Kuck, "The organization and use of parallel memories," IEEE Transactions on Computers, Vol. C-20, pp. 1566-1569, Dec. 1971.

[6] D.J.Kuck, R.A.Stokes, "The Burroughs scientific processor (B.S.P.)," IEEE Transactions on Computers, Vol. C-315, pp. 363-376, May 1982.

[7] D.H.Lawrie, C.R.Vora, "Multi-dimensional parallel access computer system," U.S. Patent n°.4,051,551 Sept. 27, 1977.

[8] D.H.Lawrie, C.R.Vora, "The prime memory system for array access," IEEE Transactions on Computers, Vol. C-31, 5, pp. 435-442, May 1982.

[9] D.H.Lawrie, "Access and alignment of data in an array processor," IEEE Transactions on Computers, Vol. C-24,12 , pp. 1145-1155, Dec. 75.

[10] T.Feng, C.Wu, "On a class of Multistage Interconnection Networks", IEEE Transactions on Computers, Vol. C-29, 8, pp.694-702, Aug. 1980.

[11] D.C.Van Voorhis, T.H.Morrin, "Memory systems for image processing," IEEE Transactions on Computers, Vol. C-27, 2, pp. 113-127, Feb. 78.

[12] D.Steinberg, "Invariant Properties of the Shuffle-Exchange and a Simplified Cost-Effective Version of the Omega Network," IEEE Transactions on Computers, Vol.C-32, 5, pp. 444-450, May 1983.

[13] C.Wu, T.Feng, "The Reverse-Exchange Interconnection Network," in Proc. 1979 Int. Conf. Parallel Process., Aug.1979, pp. 160-174.

[14] J.Schröder, U.Trottenberg, "Reduktionsverfahren für Differenzengleichungen bei Randwertaufgaben", Numer. Math. 22, pp. 37-68, 1973.

[15] A.Brandt, "Multi-level Adaptative Solutions to Boundary-value problems", Math. Comp. 31, pp. 333-390, 1977.

[16] K.Stüben, U.Trottenberg, "Multigrid methods: Fundamental Algorithms, Model Problem Analysis and Applications", Proceedings of Multigrid Methods Conference held at Köln-Porz, November 23-27 1981, Lectures Notes in Mathematics, Springer Verlag, Berlin 1982.

[17] D.Gannon, "On Mapping Non-uniform P.D.E. Structures and algorithms onto Uniform Array", International Conference on Parallel Processing, pp.100-105, Aug.1981.

[18] J.R.Van Rosendale, "Rapid Solution of Finite Element Equations on Locally Refined Grids by Multi-level Methods", Department of Computer Science, University of Illinois, UIUCDCS-R-80-1021, Urbana, Illinois 1980.

[19] D.E.Batcher, "STARAN parallel processor system hardware", Fall Joint Computer Conference, AFIPS, Conference Proceedings, vol.8, 1974.

[20] H.J.Siegel, "Interconnection Networks for SIMD machines", Computer, vol.12, pp.57-65, June 1979.

[21] J.Lenfant, "Parallel Permutations of Data: A Benes Network Control Algorithm for Frequently Used Permutations", IEEE Transactions on Computers, C-27, 7, pp.637-647, July 1978.

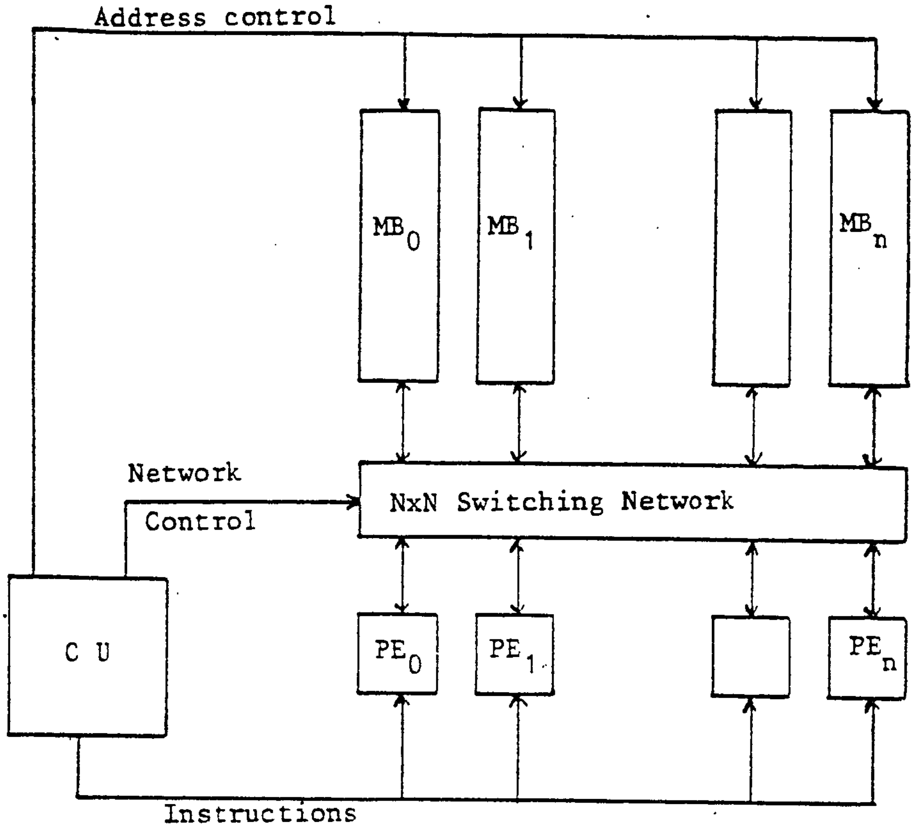


Figure 1 : An SIMD architecture.

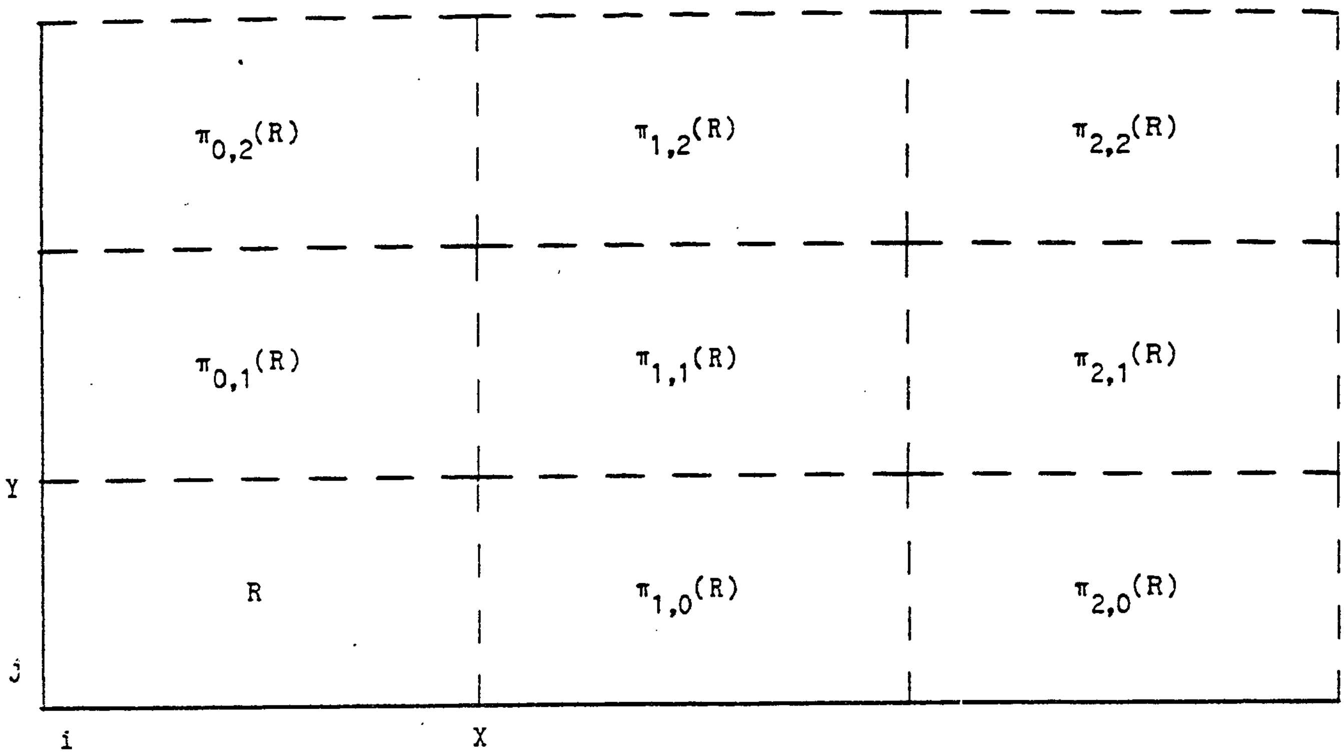


Figure 2 : Reference rectangle and reference covering.

12	13	14	15	0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15	0	1	2	3
4	5	6	7	8	9	10	11	12	13	14	15
0	1	2	3	4	5	6	7	8	9	10	11
13	12	15	14	1	0	3	2	5	4	7	6
9	8	11	10	13	12	15	14	1	0	3	2
5	4	7	6	9	8	11	10	13	12	15	14
1	0	3	2	5	4	7	6	9	8	11	10
12	13	14	15	0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15	0	1	2	3
4	5	6	7	8	9	10	11	12	13	14	15
0	1	2	3	4	5	6	7	8	9	10	11

Figure 3.a : An example of extension : the full skewing scheme.

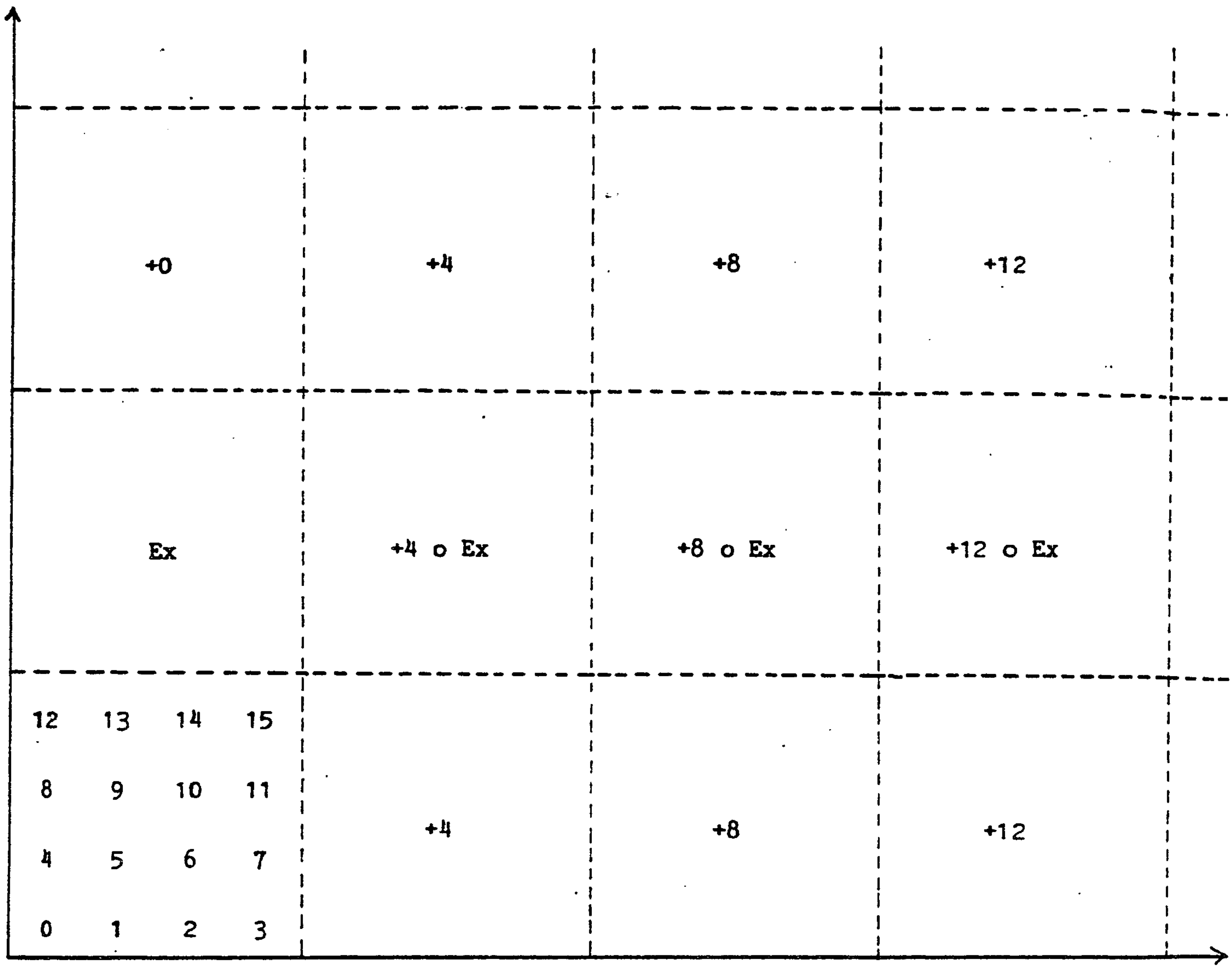
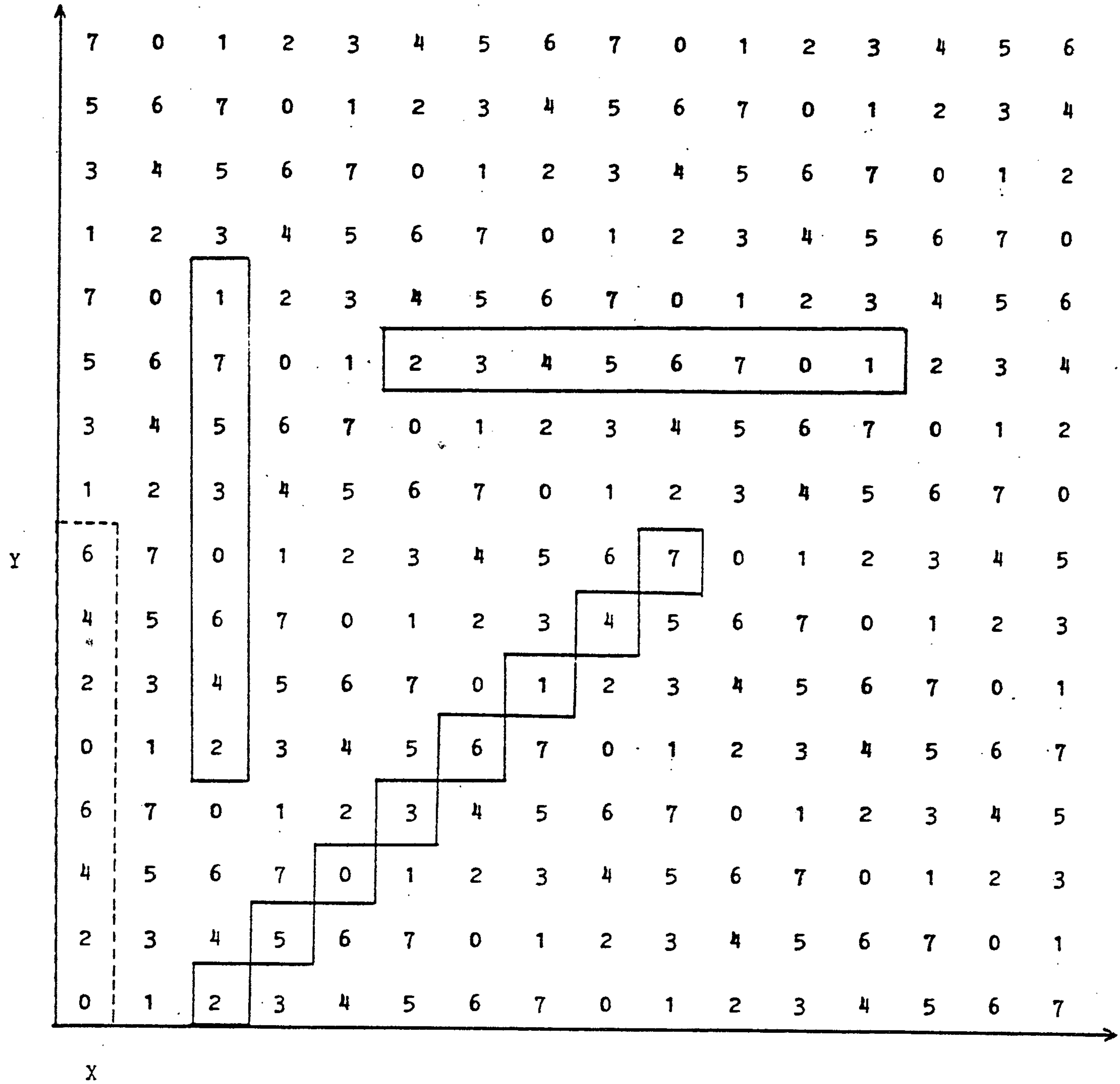


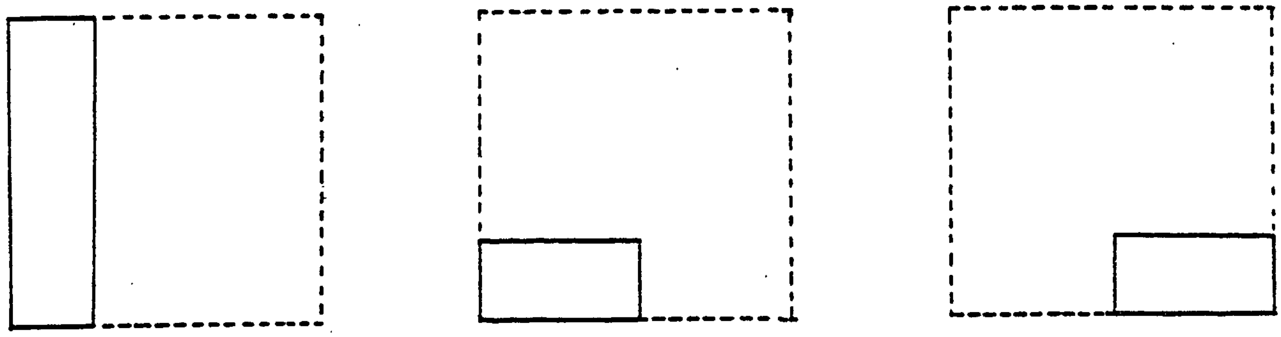
Figure 3.b : An example of extension : construction method.



<u>Figure 4</u> : Access to rows, columns and diagonals for N = 8.

<u>Legend</u> : Template compatible with ϕ.

Reference rectangle.



δ_1 δ_2 δ_3

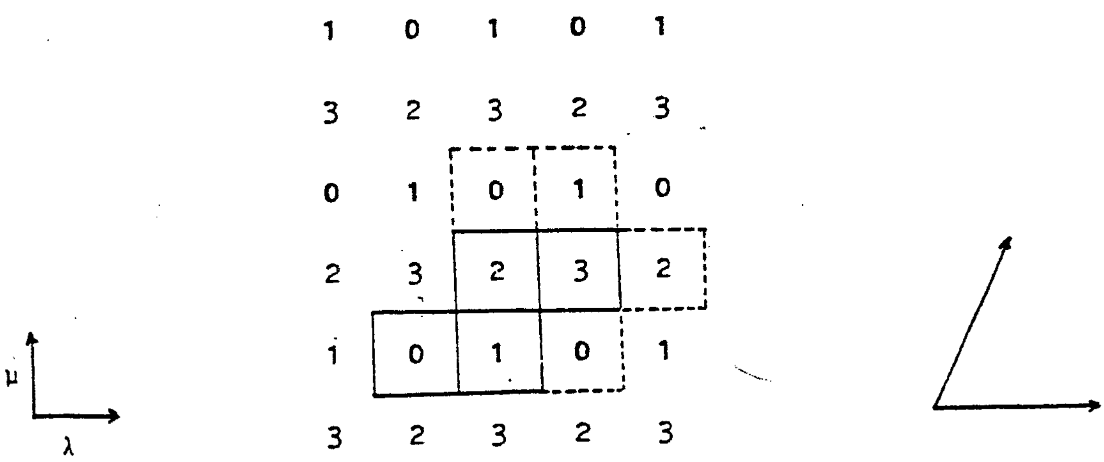
δ_1 : the cycle of 1

δ_2, δ_3: the cycles of c

$\Theta = \delta_1 \times \delta_2$ $\Theta' = \delta_1 \times \delta_3$

$\Theta, -\Theta'$: the orbits of $\hat{\lambda}$ and $\hat{\mu}$

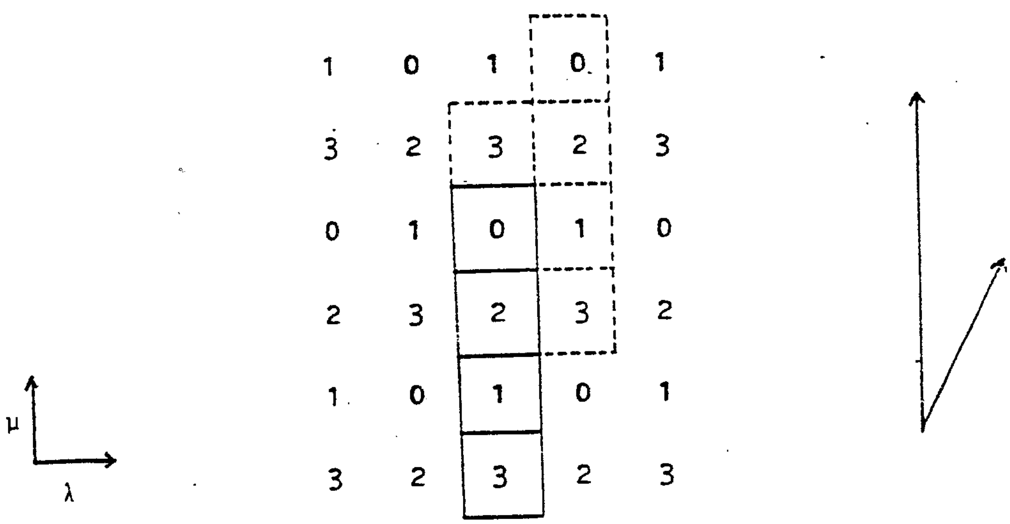
<u>**Figure 5**</u> : Orbits of a separable extension



θ_1

Horizontal basis for L_1

Figure 7.a : θ_1 with an horizontal basis.



θ_1

Vertical basis for L_1

Figure 7.b : θ_1 with a vertical basis.

Figure 7 : Two representations of the orbit θ_1 .

The base point chosen is $z = 0$.

Squares bounded with heavy lines form the parallelogram Z^2/L_1

Squares bounded with dotted lines indicate how to fold it onto a torus.

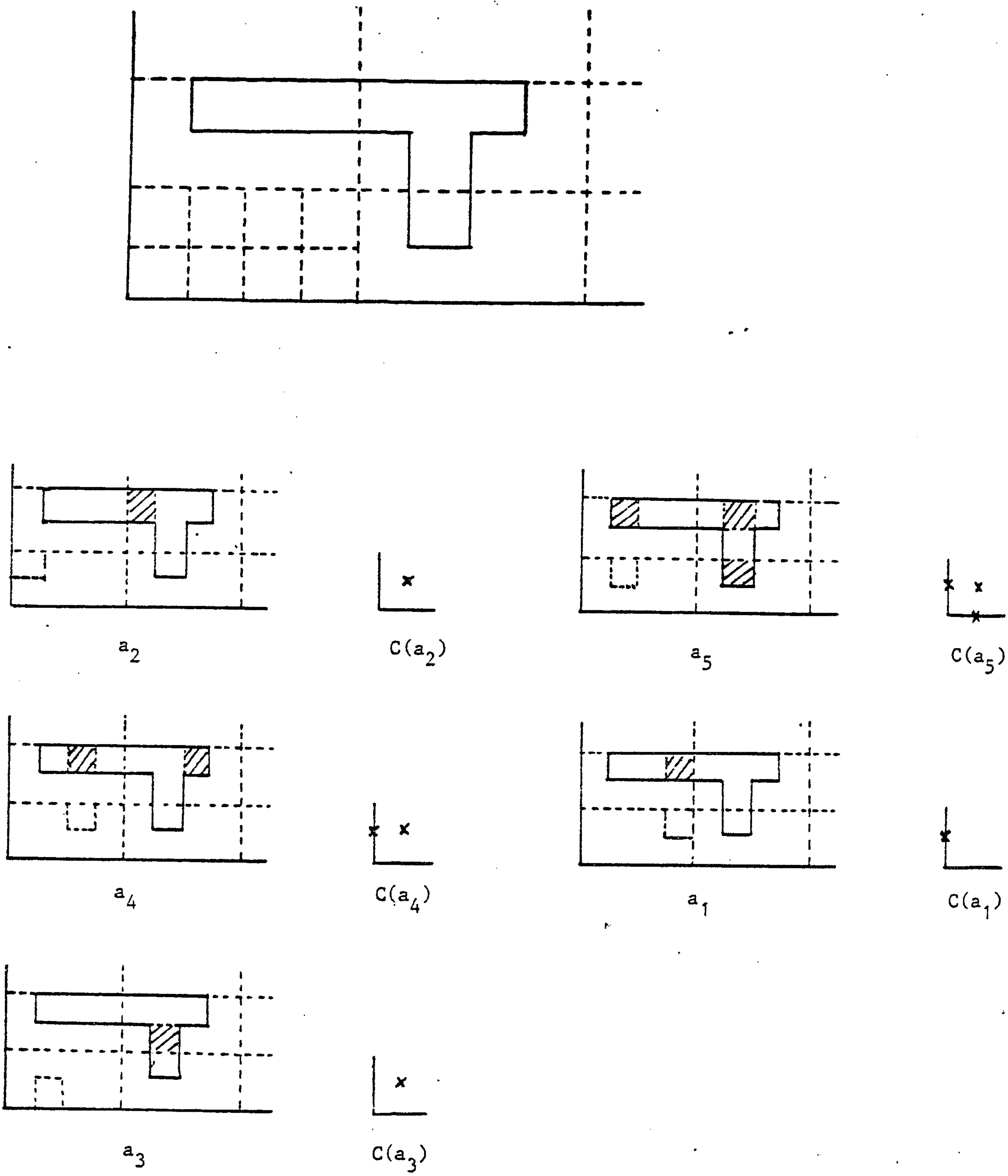
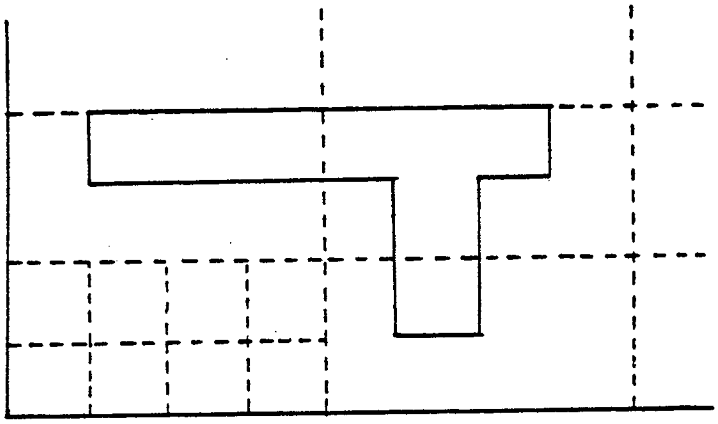
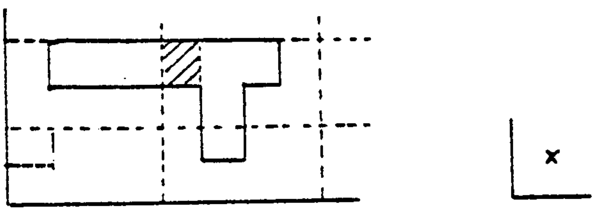
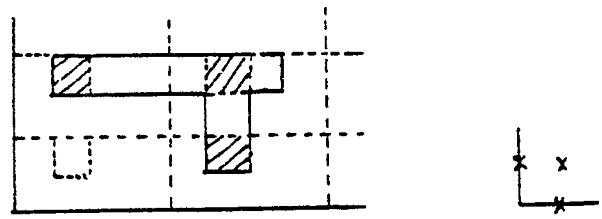
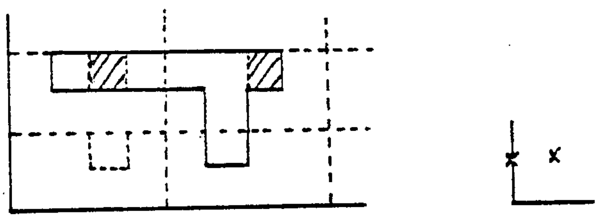


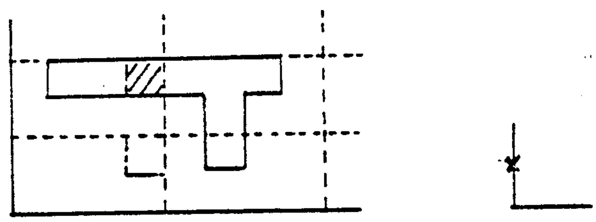
Figure 8 : An example of template decomposition.

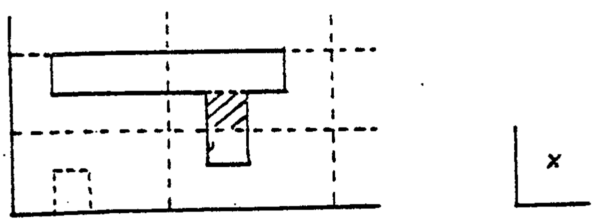


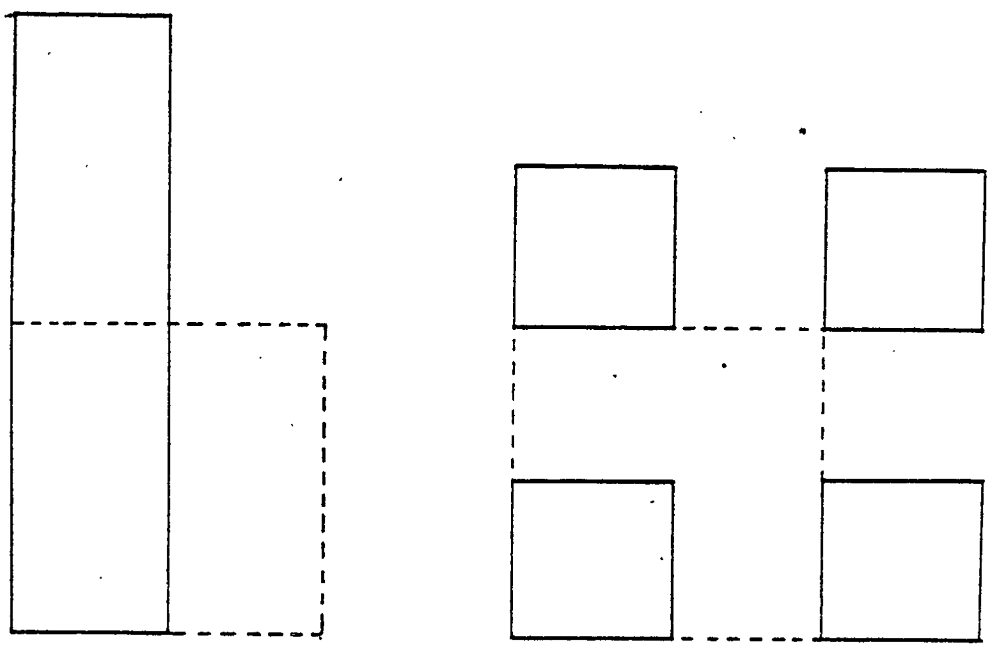








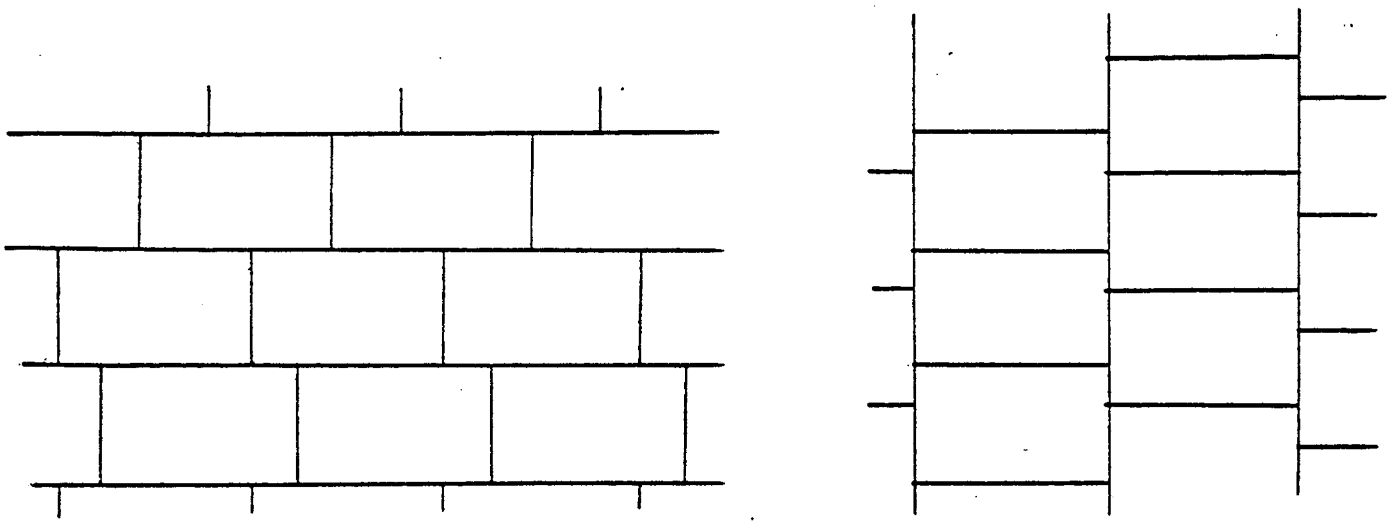




<u>**Figure 9**</u> : Two uniform templates on $R_{4,4}$.

: T_1, a one-dimensional template H = 1, K = 2

: T_2, a two-dimensional template H = 2, K = 2



Horizontal tesselation. Vertical tesselation.

Figure 10 : The two possible ways of tesselating Z^2 with a rectangle.

	4	2	6	7	4	5	0	7	4	3	6	7	4	2	0	7
	0	1	3	5	6	1	2	3	0	1	5	2	6	1	3	5
	4	5	6	7	4	3	0	7	4	2	6	7	4	5	0	7
	0	1	2	3	6	1	5	2	0	1	3	5	6	1	2	3
	4	3	6	7	4	2	0	7	4	5	6	7	4	3	0	7
	0	1	5	2	6	1	3	5	0	1	2	3	6	1	5	2
	4	2	6	7	4	5	0	7	4	3	6	7	4	2	0	7
	0	1	3	5	6	1	2	3	0	1	5	2	6	1	3	5
	4	5	6	7	4	3	0	7	4	2	6	7	4	5	0	7
	0	1	2	3	6	1	5	2	0	1	3	5	6	1	2	3
	4	3	6	7	4	2	0	7	4	5	6	7	4	3	0	7
	0	1	5	2	6	1	3	5	0	1	2	3	6	1	5	2
	4	2	6	7	4	5	0	7	4	3	6	7	4	2	0	7
	0	1	3	5	6	1	2	3	0	1	5	2	6	1	3	5
Y	4	5	6	7	4	3	0	7	4	2	6	7	4	5	0	7
	0	1	2	3	6	1	5	2	0	1	3	5	6	1	2	3
				X												

Figure 11 : A diamond scheme compatible with the template of Fig.8

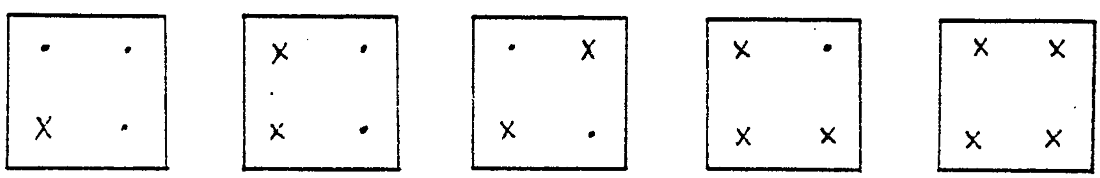


Figure 12 : The five configurations for $C(a_i)$ up to symmetries.

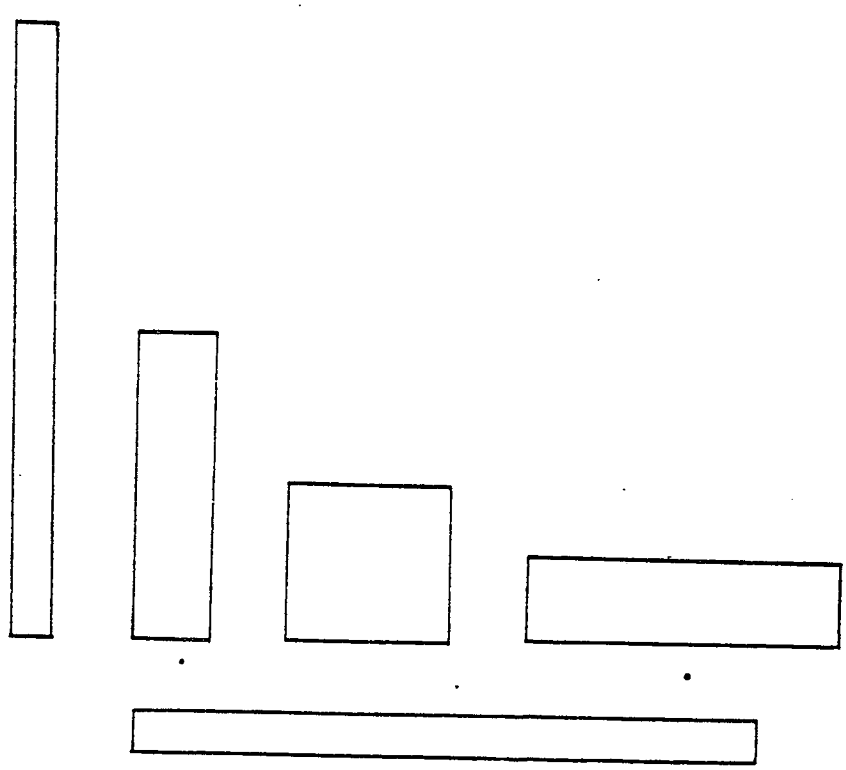


Figure 13 : Templates required for image processing.

15	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
7	8	9	10	11	12	13	14	15	0	1	2	3	4	5	6
11	12	13	14	15	0	1	2	3	4	5	6	7	8	9	10
3	4	5	6	7	8	9	10	11	12	13	14	15	0	1	2
13	14	15	0	1	2	3	4	5	6	7	8	9	10	11	12
5	6	7	8	9	10	11	12	13	14	15	0	1	2	3	4
9	10	11	12	13	14	15	0	1	2	3	4	5	6	7	8
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0
14	15	0	1	2	3	4	5	6	7	8	9	10	11	12	13
6	7	8	9	10	11	12	13	14	15	0	1	2	3	4	5
10	11	12	13	14	15	0	1	2	3	4	5	6	7	8	9
2	3	4	5	6	7	8	9	10	11	12	13	14	15	0	1
12	13	14	15	0	1	2	3	4	5	6	7	8	9	10	11
4	5	6	7	8	9	10	11	12	13	14	15	0	1	2	3
8	9	10	11	12	13	14	15	0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Figure 14 : A diamond scheme for image processing.

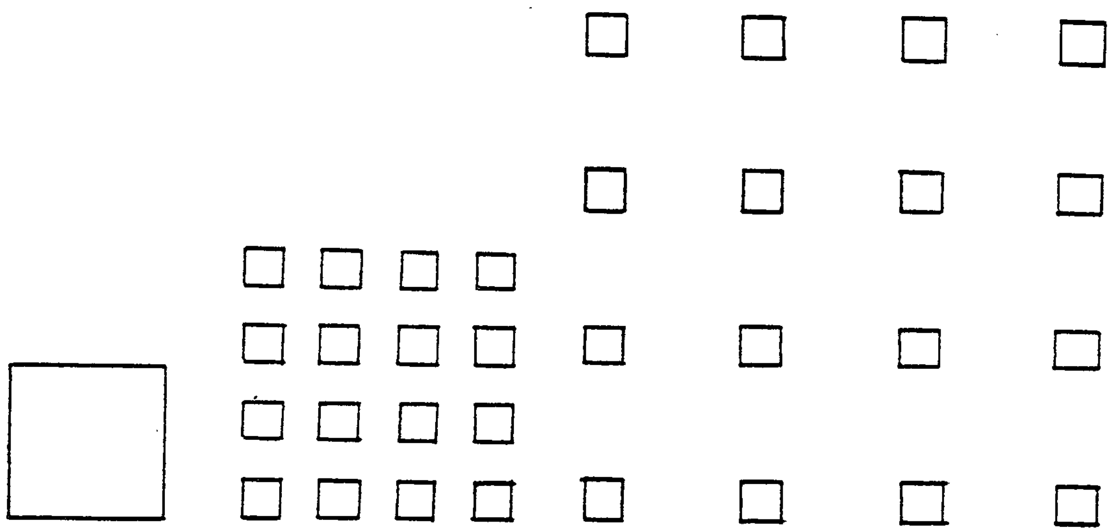


Figure 15 : Templates required for multigrid and T.R. methods.

15	3	7	11	12	0	4	8	13	1	5	9	14	2	6	10
14	2	6	10	15	3	7	11	12	0	4	8	13	1	5	9
13	1	5	9	14	2	6	10	15	3	7	11	12	0	4	8
$\boxed{12}$	0	4	8	$\boxed{13}$	1	5	9	$\boxed{14}$	2	6	10	$\boxed{15}$	3	7	11
11	15	3	7	8	12	0	4	9	13	1	5	10	14	2	6
10	14	2	6	11	15	3	7	8	12	0	4	9	13	1	5
9	13	1	5	10	14	2	6	11	15	3	7	8	12	0	4
$\boxed{8}$	12	0	4	$\boxed{9}$	13	1	5	$\boxed{10}$	14	2	6	$\boxed{11}$	15	3	7
7	11	15	3	4	8	12	0	5	9	13	1	6	10	14	2
6	$\boxed{10}$	14	$\boxed{2}$	7	$\boxed{11}$	15	$\boxed{3}$	4	8	12	0	5	9	13	1
5	9	13	1	6	10	14	2	7	11	15	3	4	8	12	0
$\boxed{4}$	$\boxed{8}$	12	$\boxed{0}$	$\boxed{5}$	$\boxed{9}$	13	$\boxed{1}$	$\boxed{6}$	14	10	2	$\boxed{7}$	11	15	3
3	7	11	15	0	4	8	12	1	5	9	13	2	6	10	14
2	$\boxed{6}$	10	$\boxed{14}$	3	$\boxed{7}$	11	$\boxed{15}$	0	4	8	12	1	5	9	13
1	5	9	13	2	6	10	14	3	7	11	15	0	4	8	12
$\boxed{0}$	$\boxed{4}$	8	$\boxed{12}$	$\boxed{1}$	$\boxed{5}$	9	$\boxed{13}$	$\boxed{2}$	6	10	14	$\boxed{3}$	7	11	15

Figure 16 : A diamond scheme for multigrid and T.R. methods.

Imprimé en France

par

l'Institut National de Recherche en Informatique et en Automatique